Apprentissage par Renforcement : Au delà des Processus Décisionnels de Markov (Vers la cognition incarnée)

Alain Dutech

NANCY UNIVERSITE

UFR mathématiques et informatique **École doctorale IAEM Lorraine**

Département de formation doctorale en informatique

Apprentissage par Renforcement : Au delà des Processus Décisionnels de Markov.

(Vers la cognition incarnée)

MÉMOIRE

présenté et soutenu publiquement le 2 décembre 2010

pour l'obtention de l'

Habilitation à Diriger des Recherches

de l'Université – Nancy II

(Spécialité Informatique)

par

Alain DUTECH

Composition du jury

Rapporteurs :	Philippe GAUSSIER	Prof. Univ. Cergy-Pontoise, ETIS
	Frédérick GARCIA	DR INRA, Toulouse, MID
	Olivier SIGAUD	Prof. Univ. Paris VI, ISIR
Examinateurs :	François CHARPILLET	DR INRIA, Nancy, LORIA
	Jean-François MARI	Prof. Univ. Nancy2, LORIA
	Manuel SAMUELIDES	Prof. Supaéro, Toulouse, CERT

Laboratoire Lorrain de Recherche en Informatique et ses Applications — UMR 7503

Loria

Mis en page avec la classe thloria.

Résumé

Ce document présente mon "projet de recherche" sur le thème de l'*embodiment* ("cognition incarnée") au croisement des sciences cognitives, de l'intelligence artificielle et de la robotique. Plus précisément, je montre comment je compte explorer la façon dont un agent, artificiel ou biologique, élabore des représentations utiles et pertinentes de son environnement.

Dans un premier temps, je positionne mes travaux en explicitant notamment les concepts de l'*embodiment* et de l'apprentissage par renforcement. Je m'attarde notamment sur la problématique de l'apprentissage par renforcement pour des tâches non-Markoviennes qui est une problématique commune aux différents travaux de recherche que j'ai menés au cours des treize dernières années dans des contextes mono et multi-agents, mais aussi robotique. L'analyse de ces travaux et de l'état de l'art du domaine me conforte dans l'idée que la principale difficulté pour l'agent est bien celle de trouver des représentations adaptées, utiles et pertinentes. J'argumente que l'on se retrouve face à une problématique fondamentale de la cognition, intimement liée aux problèmes de "l'ancrage des symboles", du "frame problem" et du fait "d'être en situation" et qu'on ne pourra y apporter des réponses que dans le cadre de l'*embodiment*.

C'est à partir de ce constat que, dans une dernière partie, j'aborde les axes et les approches que je vais suivre pour poursuivre mes travaux en développant des techniques d'apprentissage robotique qui soient incrémentales, holistiques et motivationnelles.

Mots-clés: Sciences Cognitives, Intelligence Artificielle, Robotique, Apprentissage par Renforcement, Représentations, Environnements non-Markoviens, POMDP

Mis en page avec la classe thloria.

Sommaire.

Guide de Lecture

Ce document présente mon projet de recherche en y intégrant mes travaux passés afin en vue d'obtenir une "Habilitations à Diriger des Recherches". La présentation de mes travaux passés et de mon projet de recherche n'y forment pas deux parties séparées et presques indépendantes. Au contraire, c'est bien le document dans son ensemble qui constitue mon projet de recherche et j'ai inclus mes différents travaux de recherches au fil de la présentation en fonction de leur thématique, ce qui me permet non seulement de les situer par rapport à l'existant mais aussi de les discuter dans l'optique des mes travaux futurs.

Ainsi, la section 2.5, page 22, présente succinctement l'ensemble de **mes travaux passés** et pointe sur les différentes sous-parties où ces travaux sont plus longuement détaillés. Une vue synthétique de mon **projet de recherche** se trouve en section 6, page 71 et un **CV détaillé** en annexe (page 93) offre une vue de mes différentes activités de recherche. Enfin, à partir de la page 107, on trouvera une **sélection d'articles** qui forment autant de vues plus spécifiques et plus approfondies sur mes principaux travaux.

Bonne lecture.

1

Introduction

"*Pourquoi sommes-nous intelligents ?*"

Pour étudier cette question en apparence toute simple mais qui cache en fait une multitude de questions toutes plus complexes les unes que les autres, mon activité de recherche se situe à la croisée de plusieurs domaines. Je citerai dans un ordre quelconque, l'Intelligence Artificielle, les Sciences Cognitives, la Théorie de la Décision et la Robotique. La seule mention de ces termes, bien qu'informative, ne peut en rien expliciter mes activités de recherche et les questions sur lesquelles se porte mon intérêt. Tout en expliquant comment mes travaux participent ou s'inspirent des disciplines précédentes, je vais préciser l'orientation de mes recherches, passées et futures.

Les **Sciences Cognitives**, sciences pluridisciplinaires touchant aux domaines de la psychologie, de la philosophie, des neurosciences, de la linguistique, de l'anthropologie, de l'intelligence artificielle, de la sociologie et de la biologie, ont pour objet "l'analyse scientifique moderne de l'esprit et de la connaissance sous toutes ses dimensions" (Varela, 1989). La question de fond de mon travail de recherche étant l'exploration des mécanismes de la pensée, je suis un "acteur" du domaine des sciences cognitives. Plus précisément, c'est dans le cadre théorique de l'***Embodiment***[1] que je me place. Sans entrer dans les détails qui sont exposés dans la section 2.1, l'*embodiment* postule que la cognition a pour origine les processus corporels et les interactions entre le corps et l'environnement. Le corps et le cerveau forment un système fondamentalement "non-symbolique" dont émergent un comportement intelligent, la pensée, les symboles, le langage, *etc.*

Le paradigme de l'*embodiment* est essentiellement descriptif et, bien que solidement argumenté, il nécessite de passer par l'expérimentation et la réalisation effective d'entités artificielles intelligentes pour asseoir sa validité et devenir une théorie "constructive" (Brooks, 1991). Dans l'idée de créer ou de participer à la *création* d'agents intelligents, je me suis tourné vers l'**Intelligence Artificielle** et plus précisément vers le cadre de l'**Apprentissage par Renforcement** pour y chercher à la fois un cadre constructif et un cadre théorique d'analyse.

1. J'utilise le terme "*embodiment*" faute de meilleure traduction du terme anglais, "incarnation" me semblant un peu connoté. C'est aussi une sorte d'abus de langage car il faudrait en fait parler de "*embodied cognition*".

L'apprentissage par renforcement, qui tient son inspiration des travaux sur le conditionnement opérant (Skinner, 1953) et de l'apprentissage "par essais et erreurs" (Thorndike, 1911), a pris ses lettres de noblesse au sein de l'intelligence artificielle avec les travaux de Watkins (1989) et Sutton (1996). Le problème étudié par ce domaine est celui d'un agent, une entité dotée de perception et capable d'agir, placé dans un environnement et qui doit y adapter ses comportements afin d'y être de plus en plus performant.

Des différents problèmes concernant l'intelligence artificielle, je m'intéresse donc plus particulièrement à celui qui consiste à créer des entités artificielles intelligentes. Par effet de bord, par nécessité mais aussi par envie, mon intérêt se porte sur la compréhension des mécanismes de l'intelligence humaine. Pour employer les termes de Russell et Norvig, je dirais que les deux approches de l'intelligence artificielle qui m'intéressent sont alors les "systèmes qui agissent comme des humains" et les "systèmes qui pensent comme des humains" (Russell and Norvig, 1995).

Pourtant, de manière singulière, c'est dans une autre voie inspirée de la **Théorie de la Décision** (Luce and Raiffa, 1957) et de la notion de rationalité que mes travaux de recherche pourraient logiquement être catalogués. Citant toujours Russell et Norvig, j'ai plus contribué à la recherche sur des "systèmes qui agissent *de manière rationnelle*".

En effet, le cadre théorique de l'apprentissage par renforcement, et donc de mes travaux jusqu'à présent, est celui des **Processus Décisionnels de Markov** (MDP) qui permettent à un agent d'agir de manière rationnelle en cherchant à optimiser l'espérance mathématique d'un critère numérique évoluant de manière probabiliste en fonction de la dynamique des interactions entre l'agent et son environnement (Puterman, 1994). En modélisant les interactions d'un agent avec son environnement comme un processus décisionnel de Markov, l'agent peut apprendre à se doter de comportements optimaux, et donc rationnels, en utilisant les algorithmes d'apprentissage par renforcement. Plus de détails sur l'apprentissage par renforcement et son traitement formel par le biais des processus décisionnels de Markov sont donnés en sections 2.2 et 2.3.

Tant que l'on reste dans le cadre mathématique des MDP, le problème de l'apprentissage par renforcement est théoriquement résolu et il semble donc que l'on dispose d'une méthode pour construire des agents intelligents. En fait, le cadre théorique est assez contraignant et n'est jamais respecté dès que l'on s'intéresse à des robots évoluant dans le monde réel, ce qui est pourtant nécessaire si l'on veut se placer dans le cadre de l'*embodiment*.

J'ai voulu explorer plus en détail les cas où des agents doivent évoluer dans un environnement qui leur apparaît comme **non-markovien**. Il s'agit de problèmes où, en première approximation, l'agent n'a pas assez d'information à sa disposition pour pouvoir anticiper le résultat de ses actions (cf section 2.3.2). C'est en particulier le cas quand l'agent n'au qu'une vue partielle de son environnement, un cas qui est classiquement modélisé comme un **Processus Décisionnel de Markov Partiellement Observable** (POMDP, voir section 3). Ce problème me semble en effet crucial pour progresser dans notre compréhension des mécanismes de l'intelligence et ramène à des problèmes de fond de l'*embodiment* en obligeant l'agent à se construire lui-même une représentation subjective et adaptée de son environnement en étant guidé par sa tâche, son corps, ses capacités et son environnement.

Mais dans ce contexte, les outils théoriques des MDP ne garantissent pas que les algorithmes d'apprentissage par renforcement vont permettre à l'agent de s'adapter à son environnement.

La pratique montre même que le problème posé à l'agent reste souvent insoluble. Mes principaux travaux jusque là se sont donc concentrés sur la problématique de l'apprentissage par renforcement dans un cadre non-markovien, que cela soit pour mieux en comprendre les difficultés, explorer certaines voies, apporter des solutions par le biais de techniques incrémentales ou multi-agent. La section 2.5 explicite le cheminement de mes recherches qui sont ensuite détaillées dans les sections 3 et 4.

Enfin, une partie non négligeable de mes travaux relève du domaine de la **Robotique**, ainsi que je l'explique plus longuement en section 5. Le terme "robotique" s'applique car j'ai mis à l'épreuve plusieurs algorithmes d'apprentissage par renforcement sur différentes plateformes robotiques. Cependant, il ne s'agit pas de robotique classique, plus axée sur la planification de trajectoires à l'aide d'un modèle géométrique et explicite du monde, mais plus d'une robotique considérée comme "cognitive", c'est-à-dire une robotique qui s'inspire des fonctions cognitives du cerveau [2]. En cherchant à doter des robots de capacité à apprendre à améliorer leur comportement dans des situations non-markoviennes, je me concentre moins sur les performances des robots que sur les principes à mettre en œuvre pour obtenir des robots plus adaptés, plus intelligents en quelque sorte. Mais mes travaux en ce domaine restent encore très préliminaires.

Fort de ces expériences, je compte poursuivre dans cette voie en donnant à mes travaux une tournure encore plus "incarnée" et en mettant encore l'accent sur des préoccupations plus directement en lien avec le courant de pensée de l'*embodiment*. Ainsi que le présente mon "**projet de recherche**" de la section 6, je compte principalement explorer la façon dont un robot autonome situé peut élaborer des représentations de son environnement. Pour cela, je considère qu'il doit être guidé par sa tâche et ses motivations dans un cadre d'apprentissage par renforcement au sein du paradigme de l'*embodiment*. Ainsi, au cours de mes recherches futures, je vais resserrer des liens déjà tissés avec le domaine de l'apprentissage en neurosciences computationnelles, approfondir l'utilisation de l'apprentissage par renforcement en robotique développementale et explorer les liens entre apprentissage et émergence.

2. Une description très synthétique du terme se trouve dans l'éditorial d'un numéro spécial de "*IEEE Robotics & Automation* consacré au sujet (Browne et al., 2009).

2

Positionnement scientifique

L'objectif de cette partie est de positionner mes recherches, non seulement du point de vue du contexte scientifique mais aussi du point de vue de la démarche. Le contexte est celui de l'apprentissage par renforcement dans un cadre non-markovien avec, en arrière-plan, le thème de l'*embodiment*. La démarche montre comment mes recherches, initiées par mes travaux de thèse, ont été menées de manière à avancer dans une direction cohérente avec mes objectifs à long terme, objectifs qui seront ensuite précisés et explicités en section 6.

2.1 Théorie de l'*embodiment*

La théorie ou le courant de pensée de l'*embodiment* s'intéresse aux mécanismes de l'intelligence et est transverse à plusieurs disciplines, aussi bien dans ses inspirations que dans ses influences. Ce courant postule qu'un corps est nécessaire à l'établissement de l'intelligence, que l'esprit (au sens de "*mind*") est indissociable du corps (au sens de "*body*"), que tous deux – voire tous trois si on y inclut le monde – forment un tout holistique.

Dans le domaine de l'intelligence artificielle, cette position a largement été construite en réaction à ce qui était alors le courant majoritaire de l'intelligence artificielle : le "cognitivisme" et que Haugeland, puis Brooks, appellent la "bonne vieille intelligence artificielle" ou GOFAI ("*Good Old-Fashioned Artificial Intelligence*") (Haugeland, 1985; Brooks, 1991, 1999). Le "cognitivisme" est le premier paradigme, au sens chronologique, des sciences cognitives. Il émane de travaux et de théories proposées par Chomsky, Fodor, Simon, Newell, Miller et on peut considérer que ses bases sont posées lors du "Symposium on Information Theory" organisé en Septembre 1956 au MIT. Inspiré par le développement des ordinateurs comme moyen de traitement de l'information et par les premiers succès de ce qui sera l'intelligence artificielle, le cognitivisme repose sur l'hypothèse forte que "nous sommes, au fond, nous-mêmes des ordinateurs", que "notre esprit fonctionne selon des principes de calcul" (Haugeland, 1989). Ainsi, un système cognitif, qu'il soit artificiel ou naturel, est un système formel, qui peut être incarné, et qui agit et raisonne sur des *représentations dites symboliques*. Ce paradigme est tellement séduisant qu'il aida à provoquer un

changement assez radical en sein de la psychologie, mettant de côté tout le courant béhavioriste pour jeter les jalons de la psychologie cognitive (Neisser, 1967; Lindsay and Norman, 1977).

Les thèmes de recherche cognitivistes en intelligence artificielle ont certes amené des succès importants et notoires (moteurs de recherche, jeux, représentation et traitement des connaissances, extraction de données, traduction automatique, planification) mais échouent pourtant à des tâches que nous – humains – considérons comme "très simples". Ces tâches, par exemple manipuler des objets, marcher, jouer avec un ballon, sont emblématiques d'une certaine intelligence naturelle et ont comme point commun de mettre en jeu notre corps en interaction dans le monde nous environnant. Le fond du problème, d'après le courant de pensée de l'*embodiment*, vient du fait que le cognitivisme considère le corps – notamment les capteurs et actuateurs, si l'on peut en parler ainsi – comme des dispositifs périphériques du cerveau. L'esprit, le cerveau, n'a qu'à être branché à des dispositifs de perception et d'action qui sont eux responsables de faire la passerelle avec "l'extérieur" pour que l'ensemble puisse interagir avec le monde.

De fait, quatre problèmes principaux ont été identifiés comme posant des difficultés qui paraissent insurmontables si l'on se cantonne dans le paradigme GOFAI (Pfeifer and Scheier, 2001).

- **Ancrage des symboles**. Décrit et formalisé par Harnad (1990), il s'agit pour l'agent de transformer ses perceptions (et ses actions) en représentations abstraites, en symboles, qui pourront être manipulés par le cerveau.
- **"*Frame problem*"**. En admettant que le cerveau dispose d'une représentation symbolique de son environnement, il lui reste le problème de représenter les effets des actions. Si tous les effets possibles sont représentés ou peuvent être déduits de la situation courante, le problème devient alors celui de savoir quels sont les éléments importants ou pertinents que l'agent doit prendre en compte pour éviter de s'embarquer dans un raisonnement infini car amenant rapidement à une explosion combinatoire du nombre d'éléments à analyser.
- **"Etre en situation"**. Un agent plongé dans son environnement doit pouvoir connaître sa situation, se créer une représentation de sa situation actuelle. Même s'il est capable d'ancrer ses symboles, encore faut-il se poser la question de savoir quels sont les éléments de la situation qui doivent faire partie de la représentation. Le monde est infiniment compliqué et en faire une représentation exacte et complète semble complètement hors de portée. Encore une fois, se pose la question de la pertinence des éléments à considérer pour éviter que l'agent ne passe son temps à construire sa représentation du monde.
- **L'homonculus**. Quelles que soient les réponses apportées aux précédents problèmes ou, plus généralement, à la réalisation d'une entité artificielle faisant preuve d'intelligence, ces réponses ne doivent pas dépendre d'un mécanisme "extérieur" qui, implicitement, résoudrait le problème majeur *à la place* de la solution proposée. Pour illustrer ce problème, l'image souvent employée est celle d'un petit être intelligent – l'homoncule – qui tirerait les ficelles au sein de notre cerveau, petit être qui aurait lui-même un homoncule qui, lui-même, aurait un homoncule, *etc.* La métaphore montre clairement que l'on arrive à une régression infinie. Mais elle ne doit pas faire oublier qu'il faut prêter une attention particulière à ne pas doter l'agent ou le robot de mécanismes *ad hoc* et pensés par nous, être humain, si l'on veut vraiment étudier l'origine de cette intelligence dans une entité artificielle. Ce problème est facile à comprendre et à détecter sur des cas concrets. Par exemple, une théorie de la vision artificielle qui postulerait qu'il existe un mécanisme permettant d'inspecter les images sur la rétine serait en fait vide de sens car le mécanisme sus-cité devrait faire lui-même partie de la théorie de la vision et non en être extérieur (Gregory, 1987). Mais on doit prêter à ce problème de l'homoncule une attention accrue quand on touche à des problèmes de

cognition plus abstraits.

L'*embodiment*, aussi appelé "*embodied cognitive science*" ou "*embodied embedded cognition*", est un courant plus récent des sciences cognitive qui essaie de proposer un cadre qui fait disparaître les problèmes précédents (par exemple, se passer de symboles rend caduque le problème de l'ancrage des symboles) et qui propose une voie plus prometteuse pour les résoudre mais qui pose aussi d'autres questions importantes. Le postulat de base de cette théorie est de considérer que le corps et le cerveau forment un tout indissociable qui est capable d'exhiber des comportements intelligents sans avoir recours à des symboles ou à des représentations abstraites. En partant de boucles sensori-motrices simples et très bas niveau, comme dans les véhicules de Braitenberg (1984), il est possible de doter des agents artificiels de comportements intelligents ou, tout du moins, qui paraissent intelligents à un observateur extérieur. Cette idée n'est pas nouvelle, Simon l'illustre en 1969 par le biais de la métaphore de la fourmi (voir figure 2.1) mais l'*embodiment* va plus loin.

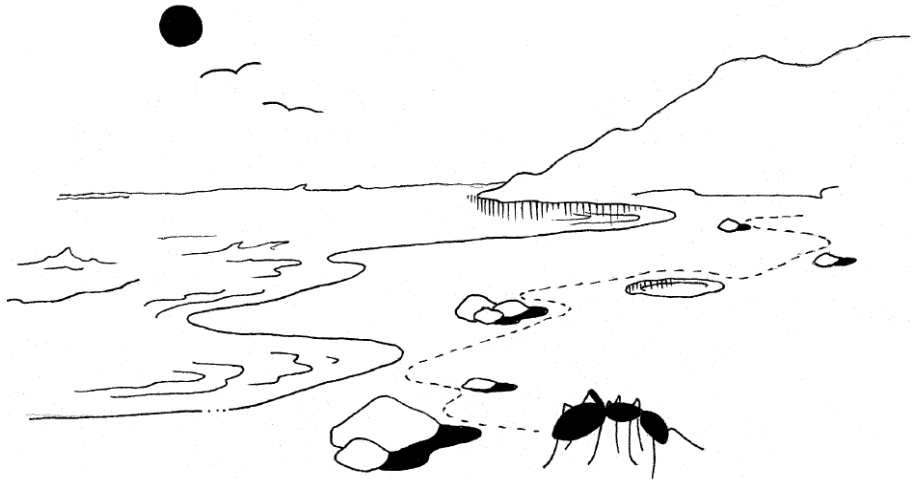


FIGURE 2.1 – **La fourmi de Simon**. Dans (Simon, 1969), pour illustrer le fait qu'un comportement simple et bas niveau peut donner lieu à un comportement qui semble complexe, l'auteur prend l'exemple d'une fourmi. Alors que la fourmi obéit à une règle très simple (*si l'antenne droite touche quelque chose, alors va à gauche et inversement, sinon avance*), elle semble en fait suivre un chemin des plus complexes en évitant les cailloux et les trous, pour se rapprocher de l'observateur. (Tiré de (Pfeifer and Scheier, 2001))

L'*embodiment* s'appuie essentiellement sur le principe d'une construction incrémentale par l'agent de représentations adaptées à son comportements, ses tâches, son environnement. En effet, bien que l'enaction sur laquelle s'appuie en partie la cognition incarnée souligne que la cognition est possible sans construction de représentations explicites, le robot peut néanmoins se créer ses représentations, mais ces représentations n'auront pas les caractéristiques classiques de celles que l'on trouve en intelligence artificielle classique. Dans un premier temps, le comportement du robot sera le fait de structure sensori-motrices préalables, assez frustres, que l'on peut qualifier d'innées, et qui permettront des associations élémentaires et favorisent certaines compétences et comportement. A ce niveau, la cognition est très dépendante du corps et peut être

très différente selon que par exemple, comme les abeilles, on perçoive les ultra-violet ou, comme l'homme, on dispose d'un appareil phonatoire complexe. Ensuite, lors de la phase d'acquisition de nouvelles compétences cognitives, ce sont des systèmes dynamiques sous formes de boucles permettant de saisir les lois des mondes intérieurs et extérieurs qui seront favorisés. Dans ce cadre, les représentations seront en fait plutôt liées à des notions d'attracteurs, de cycles limites, de frontières, *etc.* Que de ces représentations puisse ensuite émerger la capacité pour l'agent à se créer des représentations plus abstraites, plus symboliques lui permettant à terme d'élaborer un langage voire de comprendre le notre est encore purement spéculatif bien que largement espéré par la communauté de l'*embodiment*.

A mon sens, l'idée la plus intéressante de l'*embodiment* est de dire que des régularités et des invariants vont petit à petit émerger comme des abstractions ou des concepts à partir des interactions répétées de l'agent avec son environnement au travers de ses comportements sensori-moteurs. Ainsi, la notion de représentation abstraite, de symbole, loin d'être rejetée et considérée comme inutile, est au contraire conservée mais comme un des produits d'une intelligence comportementale de beaucoup plus bas niveau. Les symboles, qui ne sont pas nécessaires pour faire preuve d'intelligence, sont la clef des fonctions cognitives de haut niveau comme le langage, le raisonnement, la planification, *etc.* Mais, et en cela je rejoins une des critiques fait à ce courant de pensée, l'*embodiment* est un paradigme descriptif et il n'est ni explicatif ni constructif. Le paradigme ne dit rien des mécanismes et du substrat sous-jacent qui permettront la mise en place de comportements de bas niveau intelligents et encore moins du ou des processus par lesquels émergeront d'éventuelles abstractions. Même si cette théorie est vraie, elle ne nous dit pas comment construire des agents artificiels intelligents ou pouvant devenir intelligents. Les mécanismes de l'intelligence sont encore à découvrir, mais avec une direction de recherche différente de celle du cognitivisme.

Chaque étape du processus que nous venons de décrire rapidement amène de nombreuses problématiques ouvertes et intéressantes, et autant de questions plus spécifiques, plus limitées et plus précises. Parmi toutes ces questions, je place les questions suivantes au cœur de mon projet de recherche, ainsi qu'il sera résumé en section 6 :

- **Quelle est la part de l'inné et de l'acquis ?** Il semble utopique de pouvoir développer l'intelligence d'un agent en partant de rien. En tant qu'humains, nous avons une bonne part d'inné, des réflexes, une structure corporelle, un "précablage" du système nerveux et du cerveau, *etc.* De même, quand on travaille avec des agents artificiels, il faut trouver la bonne dose d'éléments "innés" qui permettra à l'intelligence de se développer sans pour autant escamoter le problèmes (cf le problème de l'homoncule vu précédemment).
- **Quelles sont les motivations intrinsèques d'un agent ?** Une autre facette de l'intelligence d'un agent autonome concerne le fait qu'il prenne des initiatives, qu'il soit pro-actif. On se pose ainsi la question de savoir quelles sont les motivations qui poussent l'agent à agir. Les motivations peuvent aussi être un moyen de valuer des comportements, des réactions et donc d'aider à choisir les actions pertinentes. Cette question est également à ramener à la question de l'origine, des rôles et des fonctions des émotions dans l'intelligence humaine. Dans le cognitivisme, la question des émotions a quasiment été ignorée.
- **Quels sont les mécanismes d'adaptation, de sélection d'action ?** Avant même de s'intéresser à la conceptualisation ou à l'abstraction, il reste à montrer qu'il est possible d'obtenir des comportements intelligents sans symboles. Cela nécessite d'adapter les comportements pendant la "vie" de l'agent, de choisir entre plusieurs comportements possibles et, plus globalement, d'adapter l'agent à son environnement.

– **Comment se créent ou émergent les représentations ?** Quels sont les mécanismes et le substrat de calcul qui permettent à l'agent de détecter ou de remarquer des invariants, des régularités ? Comment ces abstractions peuvent-elles être utiles ou utilisées par l'agent ?

A ces questions que l'on peut qualifier de concrètes s'ajoutent des questions plus conceptuelles, concernant le paradigme lui-même. La question essentielle est de savoir quelle(s) expérience(s) pourraient valider ou invalider le paradigme, ce qui revient à se demander s'il existe des hypothèses fondamentales de l'*embodiment* qui seraient expérimentalement vérifiables. A l'heure actuelle, la démarche suivie par les chercheurs du domaine est d'essayer de montrer par l'exemple que la théorie est valide en construisant des entités artificielles, souvent des robots, et en montrant qu'il est possible de les doter de comportements intelligents.

2.2 Apprentissage par renforcement

Le cadre général de l'apprentissage par renforcement est schématisé dans la figure 2.2. Dans ce schéma, un agent est en interaction avec un environnement, c'est-à-dire qu'il perçoit cet environnement et peut y faire des actions, ces actions pouvant modifier l'état de l'environnement. On définit le comportement de l'agent comme étant le processus par lequel il choisit d'effectuer une action, ou une suite d'actions, en fonction de sa situation actuelle.

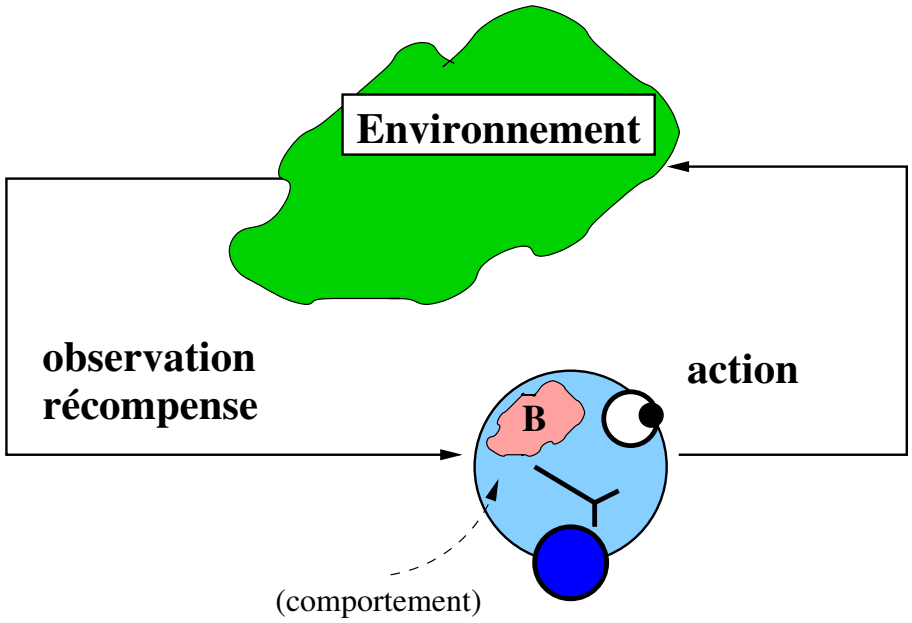


FIGURE 2.2 – **Cadre général de l'apprentissage par renforcement**

Par moment, suivant l'état de l'environnement ou suivant le résultat de certaines actions, l'agent reçoit un signal de récompense modélisé par un scalaire qui peut être positif ou négatif. Ce signal peut être vu comme une motivation ou une punition (interne ou externe) car le but de l'agent est de se comporter de manière à maximiser ses récompenses à court terme mais aussi à long terme. On parle d'apprentissage par renforcement car, petit à petit, en s'appuyant sur ses

interactions passées avec l'environnement, l'agent va essayer d'améliorer son comportement dans le but d'obtenir de plus grandes récompenses.

Inspiré par l'étude du comportement animal et notamment du conditionnement, ce cadre général de l'apprentissage par renforcement connaît un essor important dans le domaine de l'intelligence artificielle depuis le début des années 90. Il y a deux raisons principales à cela. D'une part, de nombreux problèmes de décision ou de contrôle – le système à contrôler faisant office d'environnement pour l'agent qui apprend – entrent en effet dans ce cadre général de l'apprentissage par renforcement. Les exemples affluent, allant de la robotique (Thrun et al., 2005) au backgammon (Tesauro, 1995) en passant par le contrôle d'une flottille d'ascenseurs (Crites and Barto, 1996). D'autre part, en s'appuyant sur le cadre formel des Processus Décisionnels de Markov (ou MDP), il a été possible de proposer des algorithmes permettant de résoudre de manière exacte des problèmes d'apprentissage par renforcement. La section 2.3.1 revient plus en détail sur ce formalisme et sur les algorithmes classiques de l'apprentissage par renforcement.

Les propriétés théoriques du cadre formel des MDP garantissent la convergence de certains algorithmes d'apprentissage par renforcement, mais pour autant, l'apprentissage par renforcement n'est pas une problématique résolue. En pratique, la taille des problèmes à résoudre et la complexité des algorithmes font qu'il est impossible de résoudre de manière exacte la plupart des problèmes "réalistes". Parmi ces problèmes, ceux où la *propriété de Markov* (voir section 2.3.2) n'est pas respectée me paraissent les plus importants. On peut voir ces problèmes comme des problèmes où, à l'instant de sa décision, l'agent n'a pas assez d'information sur son environnement pour être en mesure de choisir la meilleure action.

Ce genre de problème apparaît rapidement dès que l'on travaille avec des robots qui n'ont qu'une perception partielle et limitée de leur environnement. Comme nous le reverrons par la suite, c'est en particulier le cas lorsque l'on veut creuser de manière expérimentale les questions de fond de la théorie de l'*embodiment*, notamment en ce qui concerne l'élaboration par un agent de sa propre représentation du monde. Cette question, et d'autres plus en rapport avec la robotique, seront de nouveau examinées plus avant lors de l'évocation de mes travaux en robotique, dans la section 5. Et c'est cette même question qui explique mon intérêt pour l'apprentissage par renforcement pour des problèmes non-markoviens.

2.3 Un cadre formel pour l'apprentissage par renforcement

Le cadre mathématique des Processus Décisionnels de Markov (ou MDP) permet de décrire formellement la problématique de l'apprentissage par renforcement. Il permet en outre une validation théorique de certains algorithmes. On peut retenir de cette partie un peu technique que, *dans le cadre mathématique des MDP,* on peut résoudre un problème d'apprentissage par renforcement :

- en utilisant des méthodes de la programmation dynamique quand l'agent connaît le modèle du MDP (la dynamique des interactions entre l'agent et son environnement). On parle alors de *planification* ;
- en apprenant un modèle du MDP et , *simultanément*, en utilisant une des techniques précédentes, il s'agit alors d'*apprentissage par renforcement indirect* ;

– en apprenant directement un comportement optimal, sans chercher à apprendre le modèle, même quand le modèle n'est pas connu. C'est de l'*apprentissage par renforcement direct*.

00000

2.3.1 Définitions et notions essentielles

Les définitions et les concepts évoqués ici sont utiles à la compréhension du manuscrit mais ne représentent qu'une infime partie des connaissances théoriques sur les processus décisionnels de Markov. Pour approfondir ces notions, il existe une ample littérature, en particulier (Puterman, 1994), (Bertsekas and Tsitsiklis, 1996) et (Sutton, 1996). Je veux aussi mentionner l'ouvrage collectif de la communauté française sur les processus décisionnels de Markov et l'intelligence artificielle qui est entièrement consacré à ce sujet (Groupe PDMIA, 2008).

Un Processus Décisionnel de Markov (MDP) est formellement décrit par un tuple $\langle \mathcal{S}, \mathcal{A}, p, r \rangle$ où :

– $\mathcal{S}$ est un ensemble d'états discret et fini ;
– $\mathcal{A}$ est un ensemble d'actions discret et fini ;
– $p() : \mathcal{S} \times \mathcal{A} \longrightarrow \Delta(\mathcal{S})$ est une fonction de transition. $\Delta(\mathcal{S})$ est l'ensemble des distributions de probabilité sur $\mathcal{S}$. Nous utiliserons la notation $p(s'|a, s)$ pour désigner la probabilité que le processus transite vers l'état s' quand on effectue l'action a dans l'état s ;
– $r() : \mathcal{S} \times \mathcal{A} \longrightarrow \mathbb{R}$ est une fonction de récompense. Effectuer l'action a dans l'état s donne une récompense de $r(s, a)$.

Le lien avec le cadre général décrit précédemment (voir section 2.2) est assez naturel, ainsi que l'illustre la figure 2.3. Les états sont les états de l'environnement ainsi qu'ils sont perçus par l'agent et les actions sont les actions effectuées par l'agent. Les fonctions de transition et de récompense modélisent la dynamique de l'environnement et du signal de récompense qui est transmis à l'agent. Ces fonctions ne sont pas connues par l'agent qui doit néanmoins apprendre à agir au mieux, c'est-à-dire à contrôler un MDP de manière à optimiser un certain critère de la fonction de récompense.

Il existe plusieurs critères, celui classiquement utilisé est le *critère actualisé* $\sum_{t=0}^{\infty} \gamma^t r_t$ où r_t est la récompense reçue au temps t et γ un réel de $[0; 1[$. Le facteur γ est un artifice mathématique qui assure la convergence du critère mais qui peut être interprété de plusieurs façons :

– à chaque instant, le processus peut s'arrêter avec une probabilité de $1 - \gamma$;
– γ permet de pondérer l'importance des récompenses à court terme par rapport aux récompenses à long terme. Plus γ est proche de 0, moins les décisions de l'agent seront influencées par le long terme alors que si γ est proche de 1, les récompenses à long terme sont aussi importantes que celles à court terme.

On peut prouver qu'un MDP admet au moins une solution optimale qui se présente sous la forme d'une *politique* déterministe $\pi : \mathcal{S} \longrightarrow \mathcal{A}$ où l'action à effectuer ne dépend que de l'état présent du processus (voir, par exemple, (Puterman, 1994)). Cela revient à dire qu'il existe un comportement optimal pour l'agent où ce dernier n'a besoin que de ses perceptions immédiates pour décider de la meilleure action à effectuer. C'est une propriété importante car elle garantit que des agents artificiels ayant une architecture de contrôle réactive, donc très simple, suffisent pour mettre en œuvre l'apprentissage par renforcement.

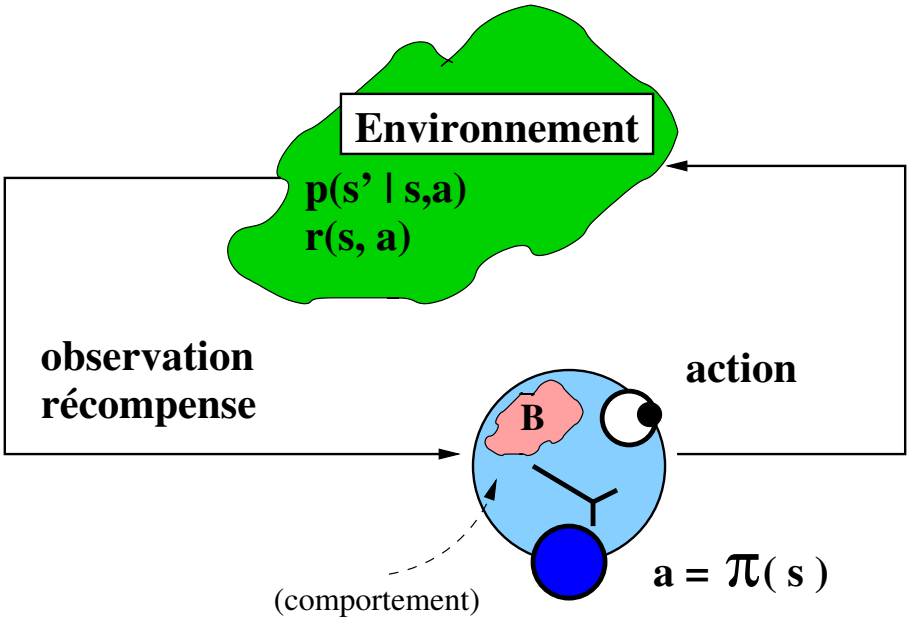


FIGURE 2.3 – **Modélisation de l'apprentissage par renforcement par un MDP.** La dynamique de l'environnement est modélisée par $p()$ et $r()$. Le comportement de l'agent est modélisé par la politique $\pi(s)$.

2.3.2 Propriété de Markov

Une des hypothèses essentielles des MDP est que le processus est *markovien*, c'est-à-dire que la connaissance du présent est suffisante pour prédire l'avenir. Formellement, le processus vérifie :

$$\Pr(s_{t+1}|a_t, s_t, a_{t-1}, \ldots, a_0, s_0) \quad = \Pr(s_{t+1}|a_t, s_t) \quad = p(s_{t+1}|a_t, s_t) \tag{2.1}$$

Cette propriété permet de tirer le meilleur parti du concept de fonction de valeur. La fonction de valeur V représente, pour un état s et une politique π donnée, l'espérance du critère de récompense quand on applique la politique et s'écrit :

$$V^{\pi}(s) \quad = \quad \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t | \pi, s_{t=0} = s\right] \tag{2.2}$$

Dans le cas de la politique optimale, la fonction de valeur optimale V^* vérifie la propriété de Bellman qui dit que la fonction de valeur en un état peut aussi se voir comme la somme de la récompense obtenue en cet état et de la fonction de valeur moyenne de l'état suivant. Plus formellement, on a :

$$V^*(s) \quad = \quad \max_{a \in \mathcal{A}} \left\{ r(s,a) + \gamma \sum_{s' \in \mathcal{S}} p(s'|a,s) V^*(s') \right\} \tag{2.3}$$

La fonction de valeur est un concept très important de la théorie des MDP car, une fois connue, elle permet de construire une politique optimale ou, en d'autres termes, un comportement optimal pour l'agent. Il a été montré qu'il existe une politique optimale qui est aussi déterministe, c'est la politique gloutonne qui, pour un état donné, utilise l'action qui va maximiser la valeur moyenne des états suivant. Cette politique optimale gloutonne se calcule aisément à partir de l'équation suivante où la politique optimale π^* se déduit de la fonction de valeur optimale par :

$$\pi^*(s) \quad = \quad \underset{a \in \mathcal{A}}{\operatorname{argmax}} \left\{ r(s,a) + \gamma \sum_{s' \in \mathcal{S}} p(s'|a,s) V^*(s') \right\} \tag{2.4}$$

En écrivant la fonction de valeur sous forme d'un vecteur $\boldsymbol{V}^*$, on définit classiquement l'opérateur $\mathcal{T}$ de la programmation dynamique à partir de l'équation (2.3) comme suit :

$$\boldsymbol{V}^* = \mathcal{T}\boldsymbol{V}^* \tag{2.5}$$

Le cadre markovien permet d'établir certaines propriétés pour l'opérateur de la programmation dynamique. Ces propriétés assurent que les MDP admettent une solution sous la forme d'une politique déterministe stationnaire, c'est-à-dire qui ne varie pas au cours du temps. Qui plus est, les algorithmes s'appuyant sur une estimation itérative de la fonction de valeur en utilisant l'opérateur de la programmation dynamique convergent de manière monotone vers la fonction de valeur optimale, seule solution de l'équation de Bellman. C'est le cas notamment de l'algorithme *Value Iteration* (Puterman, 1994) qui applique itérativement cet opérateur de la manière suivante

$$\boldsymbol{V}_i \quad = \quad \mathcal{T}\boldsymbol{V}_{i-1} \tag{2.6}$$

pour estimer la valeur optimale et en déduire une politique optimale par l'équation (2.4).

Une autre famille d'algorithmes dont le représentant le plus connu est *Policy Iteration* (Puterman, 1994) cherche directement la politique optimale en raffinant itérativement une politique pour en augmenter sa fonction de valeur.

Il est à noter que l'opérateur de la programmation dynamique définit une contraction, ce qui assure la convergence des algorithmes même dans le cas d'une mise à jour asynchrone des éléments de $\boldsymbol{V}$ ou dans le cas d'une estimation stochastique de la fonction de valeur, ce qui inclut une bonne partie des algorithmes d'apprentissage que nous allons aborder dans la section suivante.

2.3.3 Résoudre un MDP par apprentissage

Bien que l'utilisation de l'opérateur de la programmation dynamique (voir équation 2.5) nécessite la connaissance de la fonction de transition $p()$ et de la fonction de récompense $r()$, il est possible de résoudre un MDP même quand ces fonctions ne sont pas connues de l'agent apprenant. On dit alors qu'on ne connaît pas le *modèle* du MDP et l'on parle de méthodes d'*apprentissage par renforcement*, ce qui nous rapproche de notre problématique globale. Par

opposition, nous appellerons *méthodes de planification* les algorithmes évoqués précédemment et qui nécessitent la connaissance du modèle d'un MDP pour le résoudre.

Les méthodes d'apprentissage s'appuient sur des interactions répétées entre l'agent et son environnement. Ces interactions sont exploitées de trois manières différentes, formant trois catégories de méthodes d'apprentissage.

Apprentissage du modèle et de la politique

Les méthodes d'apprentissage les plus intuitives visent à apprendre le modèle du MDP ce qui permet d'appliquer, soit simultanément, soit dans un deuxième temps, les algorithmes de planification vu précédemment. Cette famille de méthode est désignée par le terme d'*apprentissage indirect* car il faut passer par le modèle pour calculer ensuite une fonction de valeur ou une politique par planification.

Les méthodes indirectes ont peu retenu l'attention des chercheurs par le passé car elles font appel à des outils déjà bien connus et étaient en général jugées peu efficientes. Il apprendre complètement le modèle pour pouvoir construire un comportement optimal. Néanmoins, ces méthodes permettent de capitaliser les informations issues des interactions avec l'environnement.

Chaque interaction de l'agent avec son environnement peut être directement mémorisée sous la forme d'un n-uplet de valeur (s_t, a_t, r_t, s_{t+1}) ou être utilisée pour apprendre les fonctions de transition et de récompense, par exemple avec des méthodes de maximum de vraisemblance. Le modèle ainsi appris peut être utilisé pour planifier les actions de l'agent mais aussi réutilisé si l'agent doit apprendre une autre tâche dans un environnement identique ou proche, on parle alors assez souvebt de transfert de connaissance [3] (Caruana et al., 1995; Caruana, 1997).

Apprentissage direct de la fonction de valeur

Les algorithmes d'apprentissage de la fonction de valeur utilisent chaque interaction pour améliorer une estimation courante d'une fonction de valeur. Il peut s'agir de la fonction de valeur de la politique courante de l'agent, comme dans les algorithmes *TD(λ)* de Sutton (1988) ou directement de la fonction de valeur optimale comme dans l'algorithme du *Q-Learning* de Watkins (1989). Dans ce dernier, après chaque interaction (s_t, a_t, r_t, s_{t+1}), l'estimation $Q(s_t, a_t)$ de la fonction de valeur optimale de l'état s_t si l'agent exécute l'action a_t est mise à jour par :

$$Q(s_t, a_t) \quad \longleftarrow \quad (1-\alpha)Q(s_t, a_t) + \alpha \left[r_t + \gamma \max_{a \in \mathcal{A}} Q(s_{t+1}, a) \right] \tag{2.7}$$

Cette famille d'algorithmes converge dans le cas des MDP sous les conditions suivantes : les couples *état-action*, en nombre fini, sont visités un nombre infini de fois et le coefficient d'apprentissage α décroît lentement vers 0 afin de vérifier que $\sum_{t=0}^{\infty} \alpha_t \rightsquigarrow \infty$ mais que $\sum_{t=0}^{\infty} \alpha_t^2 < \infty$.

3. en anglais, on trouve aussi "inductive transfer" ou "multi-task learning"

Apprentissage direct d'une politique

L'apprentissage direct d'une politique optimale, sans passer par le calcul de la fonction de valeur optimale du processus, peut s'exprimer comme un problème d'optimisation : on recherche, dans un certain espace de représentation des politiques, celle qui est la meilleure au sens du critère considéré. Les espaces couramment utilisés sont des représentations tabulaires de la politique (à chaque état est associé une action), des automates à état fini, des réseaux de neurones voire même des programmes. De cette représentation découle plusieurs méthodes pour rechercher un optimum, comme par exemple les algorithmes évolutionnaires (algorithmes et programmation génétiques), les méthodes de gradient ou, dans le pire des cas, une recherche énumérative. La thèse de Leonid Peshkin donne un aperçu plus détaillé de cette approche de l'apprentissage (Peshkin, 2002).

Les méthodes de montée de gradient, initiées par les travaux de Williams (REINFORCE, (Williams, 1992)) et popularisées et analysées par Baxter et Bartlett (Baxter and Bartlett, 2001; Baxter et al., 2001), ont été largement utilisées dans la communauté de l'apprentissage par renforcement. Elles sont en effet simples à mettre en œuvre, ont un comportement "anytime" et, comme l'ont montré Baxter et Bartlett, ces méthodes restent utilisable et donnent des résultats intéressants même dans le cas des processus qui ne sont plus markoviens. C'est dans ce cadre que nous les avons employées, comme nous y reviendrons quand nous aborderons plus en détail le cadre non-markovien et plus particulièrement au cours de nos travaux sur l'apprentissage de combinaisons de comportements (section 3.5).

2.4 Limites et problématiques actuelles de l'apprentissage par renforcement

L'apprentissage par renforcement reste un domaine de recherche des plus actifs en intelligence artificielle. Malgré des réussites intéressantes[4], de nombreux problèmes théoriques et pratiques sont à l'origine de thématiques de recherches importantes et intéressantes. Sans prétendre être exhaustif, je vais donner un aperçu des limitations les plus cruciales de l'apprentissage par renforcement en pratique. Puis, je vais m'attarder sur une hypothèse fondamentale de la théorie de l'apprentissage par renforcement que je vais ensuite m'efforcer de relâcher, à savoir le caractère markovien du processus d'interaction entre l'agent et son environnement.

2.4.1 Des algorithmes coûteux

Les propriétés théoriques des MDP permettent d'assurer la convergence de certains algorithmes d'apprentissage ou de planification vers des solutions optimales. Mais la mise en œuvre pratique de ces algorithmes pose des problèmes qui sont loin d'être résolus. Une des limitations principales concerne le coût en temps de calcul et en taille mémoire des algorithmes d'apprentis-

4. voir en particulier http ://umichrl.pbworks.com/Successes-of-Reinforcement-Learning

sage. De manière générale, la complexité en temps des algorithmes dépend du nombre d'états, du nombre d'actions et souvent du temps de mélange du processus [5].

Par exemple, l'algorithme classique du Q-Learning doit stocker $|\mathcal{S}| \times |\mathcal{A}|$ valeurs, ce qui n'est pas exorbitant sauf si la taille de $\mathcal{S}$, ou $\mathcal{A}$, l'est. Par contre, théoriquement, il faudrait que chacune de ces $|\mathcal{S}| \times |\mathcal{A}|$ valeurs soit ré-estimée un nombre infini de fois pour assurer la convergence de l'algorithme. En pratique on limite ce nombre de ré-estimations, d'autant que chacune dépend d'une interaction avec le système, interaction qui peut elle aussi être coûteuse en temps et en mémoire. Dès lors, la convergence n'est plus assurée.

De nombreux travaux cherchent donc à réduire le nombre des états ou des actions du MDP. Il est possible d'agréger les états ou les actions (Peng and Williams, 1996; Wiering and Schmidhuber, 1997; Li et al., 2006). Ainsi, pour un robot mobile, plutôt que de considérer chacun de ces états possibles, on peut abstraire le monde sous forme de pièces et de couloirs (chacun étant constitué d'un ensemble d'états) et des actions pour passer de l'un à l'autre (une action pouvant être une séquence d'actions de plus bas niveau). Actuellement, de nombreux travaux s'intéressent aux représentations factorisées des MDP pour diminuer le nombre d'états et d'actions (Boutilier et al., 1995; Boutilier, 1999; Kearns and Koller, 1999; Degris, 2007).

Une autre approche consiste à ne pas rechercher la fonction de valeur exacte de tous les états ou de tous les couples (état, action) mais de rechercher une approximation de cette fonction dans un espace plus simple (Bertsekas and Tsitsiklis, 1996; Garcia and Serre, 2000; Sutton et al., 2000; Kearns and Singh, 2002). Connaissant la valeur de certains états, des outils de régression permettent d'estimer la valeur des autres états. Outre la régression linéaire ou pondérée (Bradtke and Barto, 1996; Boyan, 1999), des outils comme les réseaux de neurones ou les méthodes des plus proches voisins ont été étudiées et expérimentées (Coulom, 2002). Ces approximations peuvent aussi s'appuyer non pas sur les états mais sur certaines caractéristiques des états [6]. Ainsi, pour le jeu de Tetris qui comporte plus de 10^{60} états, une vingtaine de caractéristiques sont suffisantes pour apprendre des fonctions de valeur permettant de "très bien" jouer (Thiery, 2007).

D'autres solutions passent par une hiérarchisation du modèle (Parr, 1998; Dietterich, 2000; Ghavamzadeh and Mahadevan, 2007). Le problème est divisé en sous-problèmes de tailles réduites, résolus ou appris dans le cadre des MDP, et un MDP global orchestre et coordonne les différents sous-MDP. Si chacun des MDP est d'une taille qui lui permet d'être résolu, le problème global peut être résolu.

Une autre approche pour résoudre les problèmes de grande taille est de passer par la recherche *en ligne* de solution. Cette approche est complémentaire des autres approches car elle peut aussi être utilisée pour raffiner ou préciser une solution approximative obtenue par une des méthodes précédentes. Le principe de la recherche de solution en ligne est de partir de l'état *actuel* du processus et de résoudre le MDP à partir de cet état seulement et pour un certain horizon.

5. "mixing time" en anglais. Etant donné une politique stationnaires, c'est le temps moyen pour que la distribution de probabilité sur les états du processus se stabilise. En effet, pour une politique donnée, dans les MDP que nous considérons, quelle que soit la probabilité initiale d'être dans un état, l'évolution du processus va lentement amener cette distribution de probabilité sur les états vers une distribution particulière qu'on appelle la distribution stationnaire.

6. On parle de "features" dans la littérature

Le principal avantage est alors de concentrer les ressources de calcul sur des états du monde qui seront effectivement utiles et visités lors de la prise de décision. Les principaux algorithmes à la base de cette approche sont les algorithmes RTDP ("*Real-Time Dynamic Programming*" (Barto et al., 1995)) et LAO* (Hansen and Zilberstein, 2001) qui est une sorte d'extension de l'algorithme A* aux graphes cycliques des MDP. Cette idée est à la base de grand succès de la planification avec des MDP, notamment en ce qui concerne le jeu de Go (Gelly and Wang, 2006; Coulom, 2007; Lee et al., 2009).

Les approches précédentes pour aborder des problèmes de grande taille cherchent principalement à montrer que les algorithmes proposés permettent de trouver des solutions qui, si elles ne sont pas optimales, s'en rapprochent autant que possible. Pour ma part, je pense que la principale difficulté consiste, pour l'agent, à découvrir et construire ces abstractions du problème. Construire ou découvrir une abstraction paraît plus facile quand le modèle est connu, ce qui ajoute encore un intérêt aux méthodes d'apprentissage indirectes. Une analyse de la structure des transitions, des trajectoires les plus vraisemblables, des états "d'étranglement" ou de "passage obligé" permet par exemple de regrouper des états en des états agrégés. Même dans ce cadre, le choix d'une représentation pertinente est délicat. Quand le modèle n'est pas connu, il est encore plus difficile pour l'agent de trouver ou découvrir une représentation lui permettant de résoudre des problèmes de grande taille par apprentissage par renforcement.

La problématique soulevée ici rejoint en fait deux des difficultés rencontrées par l'intelligence articielle classique (ou GOFAI) décrite précédemment, en section 2.1, à savoir le "*Frame problem*" et le problème "d'être en situation". Si le nombre d'états possible est trop grand, ce qui serait le cas si tous les éléments du monde étaient représentés dans l'état, il est alors difficile voire impossible d'utiliser les méthodes de l'apprentissage par renforcement sans y passer un temps infini. La solution serait de savoir quels sont les éléments importants de la situation courante pour dimininuer le nombre d'états à considérer, mais là encore il faudrait avoir déjà trouvé la solution optimale pour savoir *a posteriori* quels étaient les éléments à considérer. On peut noter que les techniques d'apprentissage en ligne se rapprochent un peu du concept d'être en situation puisqu'on porte un intérêt spécial à l'état courant, mais dans ce cas il se peut néanmoins que le nombre de transitions à considérer devienne rapidement trop important si, parmi toutes les actions possibles sur le monde, celles qui sont pertinentes n'ont pas déjà été pré-sélectionnées. On se retrouve toujours avec la même problématique à résoudre.

Je pense que l'on se retrouve ici devant un des problèmes fondamentaux de l'intelligence artificielle et, d'une certaine façon sans doute encore trop "symbolique", on retrouve une des questions principale de l'*embodiment* : Comment se créent ou émergent les représentations (voir section 2.1).

2.4.2 Un cadre théorique trop limitant ?

Les problèmes qui viennent d'être évoqués restaient dans le cadre classique et théorique de l'apprentissage par renforcement : bien que de grande taille, les problèmes étaient markoviens avec des espaces d'états et d'actions discrets. Mais les problèmes réels auxquels on voudrait s'attaquer nécessitent de considérer des cas en dehors de ce cadre confortable et en rendent la résolution de plus en plus complexe.

Le cas d'un espace d'états continu se rencontre couramment, c'est même la norme dans nombre de problèmes réalistes :"position", "distance", "vitesse" s'expriment plus souvent par des données réelles qu'entières. On peut certes définir l'équivalent de l'opérateur de Bellman dans un cadre continu [7] mais la recherche d'une politique optimale pose des problèmes théoriques et pratiques importants. Sans entrer dans des détails qui sont particulièrement bien exposés et traités dans les travaux de Rémi Munos (voir (Munos, 2004, 2006, 2007)), il faut trouver une discrétisation adaptée et pertinente de l'espace d'états ou une approximation adaptée pour pouvoir borner l'erreur entre la vraie fonction de valeur et la fonction de valeur estimée à partir des échantillons.

Etrangement, bien que le cas d'un espace d'action continu soit aussi important et courant, il a été beaucoup moins étudié que celui des états continus. Il semble possible d'appliquer les méthodes envisagées pour des états continus avec, néanmoins, au moins une difficulté supplémentaire. L'utilisation de l'opérateur de Bellman nécessite de connaître le maximum sur l'ensemble des actions de la fonction de valeur pour un état donné. Selon la complexité de la fonction de valeur, trouver un extremum peut déjà être une opération difficile mais c'est encore plus difficile étant donné que, dans le même temps, la fonction à maximiser est en train d'être estimée.

De plus, le cadre classique suppose une horloge rythmant les instants de prise de décision : les choix sont faits à des instants discrets. Mais si les actions ont des durées différentes ou qui ne sont pas connues à l'avance, le cadre classique ne convient plus. Là encore, des extensions ont été proposées, comme par exemple les processus semi-markoviens (Puterman, 1994) ou leur généralisation si plusieurs actions concurrentes tentent de contrôler le processus (Younes and Simmons, 2004; Rachelson et al., 2007; Aberdeen and Buffet, 2007). De même, la question de savoir à quel instant il est préférable d'agir est en général éludée (Cushing et al., 2007).

Ces trois aspects du cadre classique de l'apprentissage par renforcement sont importants à garder à l'esprit car dans le cadre de l'*embodiment* nous allons chercher à travailler sans symboles. Considérer un ensemble d'état, ou d'action, discret et fini est typiquement une utilisation implicite de symboles. Pour échapper au symbole, une solution est de se tourner vers des environnement plus numériques, plus continus, avec les problèmes que l'on vient d'évoquer.

Tous les problèmes considérés jusqu'à présent faisaient l'hypothèse de modèles stationnaires, c'est-à-dire des modèles où les probabilités de transition et la fonction de récompense ne dépendaient pas du temps. Quand ce n'est pas le cas, on parle de problèmes non-stationnaires. Théoriquement, ce problème n'en est pas vraiment un car il suffit de considérer le temps comme un des élements de l'état du système pour résoudre ce problème comme un MDP, ce qui n'est pas vraiment une solution en pratique.

Quand *plusieurs* agents agissent et apprennent indépendamment dans l'environnement, de nouveaux problèmes apparaissent. J'aborde ce cadre de l'apprentissage par renforcement dans les systèmes multi-agents plus en détail dans la section 4 car, outre des problèmes de coordination dus au fait que les agents peuvent chacun trouver une politique globale optimale différente, les problèmes multi-agents placent chaque agent dans un cadre non-stationnaire et non-markovien.

7. opérateur de Hamilton-Jacobi-Bellman, voir (Bellman, 1957)

2.4.3 Au delà de Markov

Apprendre par renforcement quand l'environnement perçu par l'agent ne respecte pas l'hypothèse de Markov est très délicat. C'est encore un problème largement ouvert. Pourtant, ce cadre est des plus fréquents car la perte du caractère markovien peut avoir plusieurs causes assez courantes.

- Les perceptions immédiates de l'agent ne lui permettent pas de connaître l'état complet du système qu'il forme avec son environnement. Par exemple, un agent qui doit mettre un objet dans une boîte pour en faire ensuite un paquet cadeau et qui se trouve devant une boîte fermée ne sait pas si l'objet est déjà à l'intérieur ou pas. Sans mémoire, il ne peut résoudre ce dilemme en utilisant l'apprentissage par renforcement classique.
- La discrétisation d'un état continu implique souvent que l'agent se retrouve face à un processus non-markovien. Comme l'illustre la figure 2.4, le mécanisme de discrétisation peut avoir pour conséquence que l'effet d'une action n'est pas indépendante du passé récent du processus. C'est une forme d'observabilité partielle.
- Plusieurs agents sont présents dans l'environnement. Si les agents ont une perception limitée de l'environnement, ce dernier peut leur apparaître non-markovien car les transitions peuvent dépendre des actions de tous les agents. Ainsi, du point de vue d'un agent, son environnement n'est pas markovien.
- Les récompenses délivrées à l'agent ne suivent pas une loi markovienne. Par exemple, un agent qui est récompensé seulement la première fois où il arrive dans un certain lieu est plus facile à représenter avec une fonction de récompense non-markovienne. Les rares travaux s'intéressant à la question se font dans un cadre où le modèle de la dynamique est connu et cherchent à construire un nouveau problème qui est markovien (Thiébaux et al., 2006).

Il est toujours possible de trouver une modélisation d'un problème pour qu'il soit markovien, mais au détriment de la taille des espaces d'états et d'actions, ce qui est vite rédhibitoire. De plus, le cadre qui m'intéresse avec l'*embodiment* en arrière plan, est celui de l'agent situé autonome, c'est-à-dire que les états du MDP doivent correspondre à des états subjectifs de l'agent. L'idée est que l'apprentissage par renforcement est mis en œuvre *par* l'agent et l'état devient alors l'*état tel que le conçoit l'agent*. Dès lors, deux options sont possibles. On peut rendre markovien le problème posé à l'agent, par exemple en le dotant de capacités de perception et de mémoire interne *ad-hoc*. Vouloir ainsi occulter le problème de cette manière s'apparente à adapter l'environnement à l'agent, ce qui est la démarche inverse de celle qui vise à rendre les agents autonomes. L'autre solution consiste à laisser l'agent s'adapter de lui-même à ce problème non-markovien. C'est pour cette raison qu'il me semble important d'étudier l'apprentissage par renforcement pour des systèmes non-markoviens afin de proposer des solutions et des algorithmes dont la mise en œuvre ne viole pas la contrainte d'autonomie de l'agent, toujours dans le but de progresser dans la compréhension des mécanismes de l'intelligence.

La suite de ce manuscrit, et notamment les sections 3, 4 et 5, est consacrée à cette problématique d'apprentissage par renforcement quand l'agent est face à un environnement non-markovien. J'y explicite les difficultés qu'il faut résoudre et les contributions que j'ai apportées au domaine. Nous y verrons aussi que, comme dans le cas markovien, le problème fondamental qui se dessine est encore celui de "l'apprentissage" d'une représentation pertinente du monde.

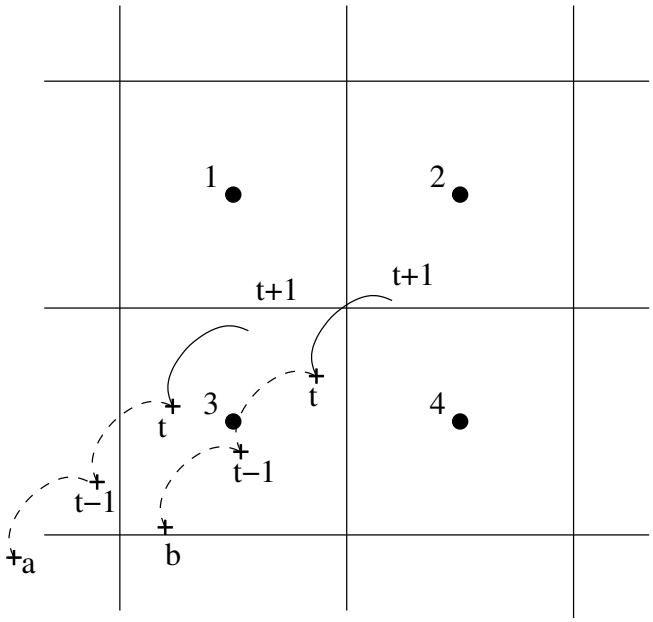


FIGURE 2.4 – **Transitions non-markoviennes dans un espace discrétisé.** Les états discrets $(1, 2, 3, 4)$ représentent chacun une zone de l'espace. Pour une même séquence d'actions et un même état (3) au temps t, suivant l'état de départ a ou b, les distributions de probabilité sur l'état discret suivant sont différentes.

2.5 Mon cheminement jusqu'à présent

Le contexte de mes recherches ayant été posé, je présente maintenant une vue très synthétique de mes travaux passés par le biais de ce que j'appelle mon "cheminement". Ce cheminement suivi au cours de mes travaux de recherche n'a évidemment de chemin que le nom puisque qu'il emprunte plusieurs routes en même temps ou utilise parfois une sorte de téléportation. La difficulté réside seulement dans le fait de devoir le présenter de manière linéaire, comme je m'y essaie maintenant.

Thèse sur l'apprentissage dans les POMDP. Au cours de ma thèse au CERT de Toulouse, sous la direction de Manuel Samuelides et soutenue en 1999, j'ai travaillé sur l'apprentissage dans une extension des processus décisionnels de Markov dans le cas où les agents n'ont qu'une connaissance et une perception imparfaite de leur environnement : les Processus Décisionnels de Markov *Partiellement Observables* (POMDP). Comme je le détaille en section 3.4, ce cadre place les agents dans un environnement subjectif qui n'est plus markovien.

Lors de cette thèse, j'ai essentiellement proposé un algorithme d'apprentissage pour les POMDP, ce qui reste très rare même à l'heure actuelle, mais dont la preuve de convergence ne concerne en fait qu'un nombre infime de problèmes, comme je l'ai montré par la suite. Le cadre des POMDP devient dès lors le cadre principal de mes travaux.
Ces travaux ont été publiés dans mon manuscrit (Dutech, 1999), lors d'un Workshop (Dutech

and Samuelides, 1999) et dans un article de la revue RIA (Dutech and Samuelides, 2003), article inclus dans ce document à la page 107.

Post-Doc sur la planification dans les POMDP. A mon arrivée dans l'équipe MAIA au Loria en 1999, j'ai analysé comment mon algorithme d'apprentissage se comportait quand les agents avaient assez de connaissance sur leur environnement pour faire de la planification. C'est ainsi que nous avons compris et montré que les hypothèses nécessaires à l'utilisation de mon algorithme étaient très restrictives et qu'elles plaidaient pour l'ajout de contraintes sur la forme des politiques optimales recherchées (cf. section 3.4).
Ces travaux ont été présentés lors de la conférence ECAI'00 (Dutech, 2000).

Détection de caractéristiques pertinentes pour les états passés. Toujours dans le cadre des POMDP, j'ai exploré avec Bruno Scherrer, alors doctorant dans l'équipe MAIA, la possibilité de détecter les états "*pertinents*" d'un processus non-markovien, c'est-à-dire les états *passés* qui, s'ils sont mémorisés par l'agent, lui permettent de redonner un caractère markovien à un processus qui ne l'est pas si on ne considère que la perception des états *présents*. Ces travaux préliminaires ont ouvert de nombreuses pistes à explorer, notamment par le fait qu'ils nécessitent que l'agent connaisse la dynamique de son environnement, ce qui est généralement difficile à apprendre (cf. section 3.3).
Ces travaux ont été présenté lors du workshop EWRL'01 (Dutech and Scherrer, 2001).

Apprentissage incrémental. Pour apprendre dans les POMDP, j'ai exploré d'autres approches, parfois plus expérimentales. Ainsi, comme le montre la section 3.5, l'approche poursuivie pendant la thèse d'Olivier Buffet qu'il a soutenue en 2003 était assez originale à l'époque car elle proposait d'ajouter à l'apprentissage par renforcement des éléments issus d'une approche développementale de l'intelligence. En combinant des comportements très basiques spécifiés et appris par l'agent, ce dernier construit des comportements plus complexes. Les nouveaux comportements ainsi obtenus peuvent eux-mêmes être combinés, et ainsi de suite, pour construire des comportements de plus en plus élaborés. Validé expérimentalement, ce travail est emblématique de la notion d'apprentissage incrémental que j'ai aussi étudié dans un cadre multi-agent et que je compte poursuivre au cours de mes recherches futures.
Ces travaux ont notamment été présentés aux conférences AAMAS'02 (Buffet et al., 2002b), ECAI'02 (Buffet et al., 2002a), AAMAS'03 (Buffet et al., 2003), SAB'04 (Buffet et al., 2004) et publié dans la revue RIA (Buffet et al., 2005, 2006). L'article (Buffet et al., 2005) se trouve aussi à la page 139 de ce document.

Apprentissage guidé et façonné. Une des concrétisations de ma période "apprentissage expérimental" se retrouve aussi lors du stage de DEA d'Olivier Buffet dans un cadre multi-agent. Par nature, les systèmes multi-agents définissent des problèmes où les agents n'ont pas accès à l'état du système, nous sommes donc toujours bien dans le cadre des POMDP. Pour être exact, nous nous trouvons même dans un cas plus difficile, celui des POMDP décentralisés.

Nous avons proposé un algorithme qui s'appuie sur la notion de *shaping*[8] pour qu'un agent confronté à un problème non-markovien puisse apprendre plus facilement une politique markovienne dont les performances soient néanmoins intéressantes (voir section 4.4). Le concept clef est de guider l'agent en lui proposant des situations d'abord très simples puis de plus en plus complexes, en faisant en sorte que ce qu'il apprend soit ensuite réutilisé dans les cas complexes. Cette démarche fait que le problème auquel est confronté l'agent est non-stationnaire en plus d'être non-markovien mais nos expériences ont montré la validité et la pertinence de l'approche. Ce concept de l'accompagnement de l'apprentissage sera aussi au cœur de mon projet de recherche. Ces travaux ont été présentés aux conférences AA'01 (Buffet et al., 2001), IJCAI'01 (Dutech et al., 2001) et publiés dans la revue JAAMAS (Buffet et al., 2007), article qui est reproduit dans ce document à la page 245.

Contrat de confiance. Ces travaux précédents sur les systèmes multi-agent ont mis en évidence des difficultés et des problèmes liés à un manque de coordination des agents. Les comportements des agents étaient limités car ils ne réagissaient qu'à leur perception immédiate de leur environnement. Avec Raghav Aras, nous avons voulu déterminer quelles informations supplémentaires devaient se communiquer les agents pour se coordonner. L'idée, ainsi qu'elle est développée à la section 4.3, est de donner aux agents la possibilité de se créer des contrats de confiance virtuels. Les problèmes et questions révélés par ces travaux intéressant rejoignent les principales limites de l'intelligence artificielle classique aussi, dans un premier temps, je ne pense pas poursuivre dans cette direction de recherche.
Ces travaux ont été présentés aux conférences AISTA'04 (Aras et al., 2004), AAMAS'05 (Aras et al., 2005), CAp'06 (Aras et al., 2006), CARI'06 (Dutech et al., 2006) inclus page 237.

La programmation mathématique à la rescousse ? La thématique des travaux de thèse de Raghav Aras que j'ai encadrée et qu'il a soutenue en octobre 2008 avait pour but de mieux comprendre et analyser les POMDP décentralisés (Dec-POMDP) qui sont un cadre assez naturel pour la modélisation des systèmes multi-agents. Nous avons pour cela puisé du côté des mathématiques et, plus précisément, du côté de la Théorie des Jeux (Osborne and Rubinstein, 1994) et de la Programmation Mathématique (Dantzig, 1991, 1998). En reformulant les problèmes comme des programmes mathématiques, nous avons acquis une meilleure connaissance de la structure des solutions des Dec-POMDP et proposé des algorithmes originaux pour les résoudre (cf. section 4.2). Actuellement, ces travaux ont un aspect pratique limité car ils ne sont applicables que sur des problèmes jouets mais ils ouvrent la porte à une nouvelle voie de recherche pour résoudre les Dec-POMDP de manière approchée. Par contre, leur extension à des problèmes d'apprentissage est clairement hors de notre portée pour l'instant.
Ces travaux ont été présentés aux conférences JFPDA'07 (Aras et al., 2007b) inclus à la page 169, ICAPS'07 (Aras et al., 2007a) et sont en soumissions, en deuxième lecture, pour la revue JAIR (la version actuelle de l'article fait l'objet d'un rapport technique (Aras and Dutech, 2009)).

Apprentissage par Renforcement et Robotique. En parallèle à ces recherches, j'ai mené une activité en robotique dans le but de porter les avancées réalisées sur des problèmes théoriques

8. Ce terme anglais, issu de la psychologie du développement, pourrait être traduit par "façonnage", mais ce n'est guère élégant.

et académiques sur des agents concrets et réels. Après plusieurs expérimentations sur des plateformes assez différentes et avec des résultats variés (voir section 5), nous arrivons maintenant à une maturité matérielle, logicielle et organisationnelle qui va nous permettre d'apporter des contributions plus significatives dans le domaine de la robotique cognitive. En ce qui me concerne, le passage à la robotique est une condition nécessaire pour mon projet de recherche qui est fortement marqué par les idées issues de la théorie de l'*embodiment* ainsi que je le détaille en section 6.
Ces travaux ont principalement fait l'objet de rapport de Master (Deflandre, 2006; Beaufort, 2009).

2.6 Pour résumer la problématique

J'ai d'abord présenté le courant de pensée de l'*embodiment* qui formule des hypothèses et des théories concernant les mécanisme de l'intelligence, le tout pouvant se résumer en une phrase : "*l'intelligence nécessite un corps*". Comme la théorie de l'*embodiment* est principalement descriptive, une des seules façons de "prouver" sa validité consiste à utiliser une démarche synthétique afin de montrer qu'il est possible de construire des entités intelligentes dans ce contexte.

C'est le cadre de l'apprentissage par renforcement que j'ai choisi pour disposer d'une théorie constructive de comportements intelligent pour des agents artificiels. Un des avantages de ce cadre est de pouvoir s'appuyer sur la théorie des Processus Décisionnels de Markov, ce qui permet d'analyser et valider les algorithmes utilisés. Mais, plus encore que les problèmes posés par la mise en œuvre de l'apprentissage par renforcement dans un cadre réel, c'est l'une des hypothèses fondamentales sur laquelle repose toute cette théorie qui m'a interpellé. L'apprentissage par renforcement n'est théoriquement validé que lorsque le problème posé à l'agent est markovien, ce qui est en général confirmé par l'expérimentation.

Or, pour de nombreuses raisons, les principes de l'*embodiment* amènent à devoir considérer l'apprentissage par renforcement dans un cadre non-markovien. C'est aussi le cas quand on considère plusieurs agents ou des robots qui ne peuvent "tricher" avec leur environnement. Mes travaux de recherche se concentrent donc sur les problématiques et les questions de l'apprentissage par renforcement dans un cadre non-markovien. En cherchant à construire des agents ou des robots qui apprennent à se comporter "intelligemment" dans des situations où la propriété de Markov n'est pas respectée, j'aspire en fait à étudier comment un agent se crée des *représentations* utiles et pertinentes pour lui. Ce faisant, j'espère mieux comprendre ce problème fondamental de la cognition et ainsi explorer les mécanismes sous-jacents de l'intelligence.

3

Apprendre dans les POMDP

Le cadre formel classique et général pour étudier l'apprentissage par renforcement dans des environnements non-markoviens est celui des Processus Décisionnel de Markov Partiellement Observables (POMDP). Nous allons d'abord donner les détails de ce formalisme dans le but de montrer l'importance de la représentation que se fait l'agent apprenant de sa tâche. Ensuite, nous aborderons le problème de l'apprentissage dans les POMDP, notamment par le biais de mes différentes contributions au domaine. Enfin, à l'issue de ce chapitre, je dresserai un bilan de ce qui a été accompli et des questions soulevées.

3.1 Formalisme des POMDP

Le concept essentiel des Processus Décisionnel de Markov Partiellement Observables (POMDP) est de considérer un système qui est fondamentalement markovien[9] mais où l'état du système n'est pas accessible à l'agent qui cherche à le contrôler ou à apprendre à le contrôler. On dit que l'agent n'a accès qu'à une observation de l'état du système. Le cadre est illustré par la figure 3.1.

Nous allons donner ici les définitions et les concepts sur les POMDP qui sont nécessaires à la compréhension de ce manuscrit. Il existe une abondante littérature sur le sujet que je vous invite à consulter pour de plus amples détails, notamment (Cassandra, 1998) et (Kaelbling et al., 1998). Enfin, encore une fois, je vous renvoie au livre collectif qui a été publié récemment (Groupe PDMIA, 2008) et qui consacre un chapitre entier aux POMDP, chapitre que j'ai co-écrit avec Bruno Scherrer.

A l'image d'un MDP (voir section 2.3.1), un Processus Décisionnel de Markov Partiellement Observable (POMDP) est formellement défini par un tuple $\langle \mathcal{S}, \mathcal{A}, \Omega, p, O, r, b_0 \rangle$ où :

- $\mathcal{S}$ est un ensemble d'états discret et fini ;
- $\mathcal{A}$ est un ensemble d'actions discret et fini ;
- Ω est un ensemble d'observations discret et fini ;

9. cette hypothèse à elle seule pourrait donner lieu à un vaste débat

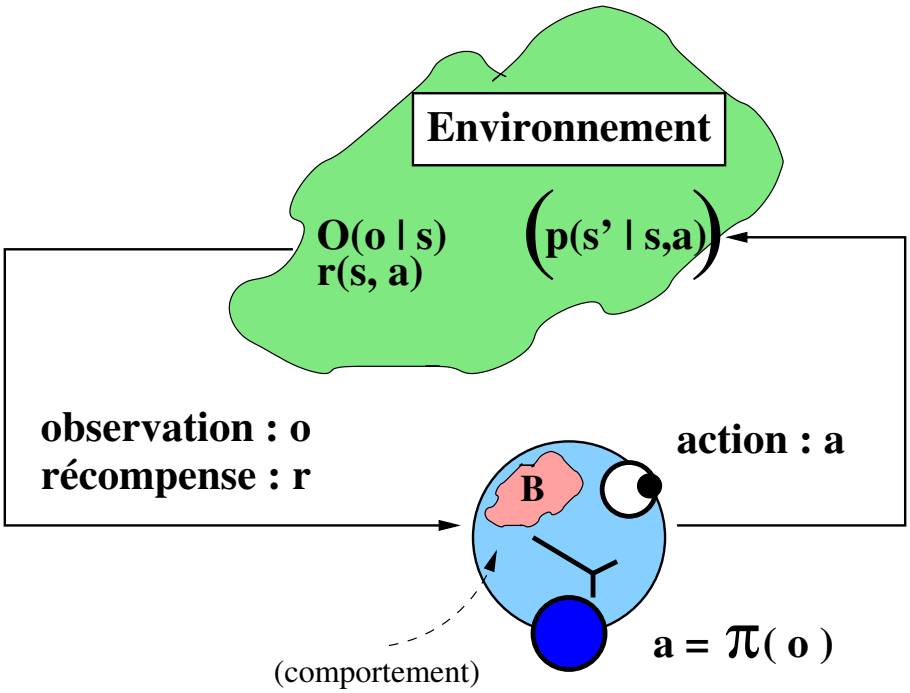


FIGURE 3.1 – **Vue schématique d'un POMDP.** La dynamique "cachée" de l'environnement est modélisée par la fonction de transition $p()$. L'agent ne peut connaître l'état s de l'environnement, il n'a accès qu'à une observation p au travers de la fonction d'observation $O()$. Il reçoit aussi une récompense r. Son objectif est alors de trouver une politique π lui permettant de maximiser une fonction des récompenses reçues. Le problème le plus crucial est de savoir sur quel espace l'agent va définir cette politique ou, autrement dit, quelle représentation de son environnement il va se construire. En effet, dans le cas général une politique définie sur l'espace Ω des observations, comme indiqué sur la figure, ne pourra jamais être optimale.

- $O()\ : \mathcal{S} \longrightarrow \Delta(\Omega)$ est une fonction d'observation. Ainsi, $O(o|s)$ est la probabilité d'observer l'observation o dans l'état s ;
- $p()\ : \mathcal{S} \times \mathcal{A} \longrightarrow \Delta(\mathcal{S})$ est une fonction de transition. Ainsi, $p(s'|a, s)$ est la probabilité que le processus transite vers l'état s' quand on effectue l'action a dans l'état s ;
- $r()\ : \mathcal{S} \times \mathcal{A} \longrightarrow \mathbb{R}$ est une fonction de récompense. Effectuer l'action a dans l'état s donne une récompense de $r(s, a)$;
- b_0 est la distribution de probabilité initiale sur les états.

Les seules nouveautés par rapport à un MDP sont donc la distribution initiale des états du système b_0, l'espace des observations Ω et la fonction d'observation associée $O()$. Les critères de performance utilisés sont les mêmes que dans le cas des MDP et nous ne considérons ici que le critère actualisé $\sum_{t=0}^{\infty} \gamma^t r_t$.

Il est à noter que, même quand la fonction d'observation $O()$ est déterministe, c'est-à-dire qu'à chaque état est associé une et une seule observation, l'agent peut ne pas connaître l'état : deux états peuvent être associés à une même observation. C'est d'ailleurs quand il y a ambiguïté sur l'état qu'il est intéressant de parler de POMDP, car sinon le problème que l'agent doit résoudre

possède la structure d'un MDP classique.

L'agent n'a accès qu'à la séquence des observations. Or, dans le cas général, cette séquence ne respecte pas la propriété de Markov (comme l'illustre entre autres la figure 3.2). On dit que l'agent est confronté à un **problème non-markovien**. Le problème *crucial* qui se pose alors est de savoir sur quel ensemble il faut définir une politique car, ainsi que nous le verrons plus loin (section 3.2), une politique s'appuyant sur la seule observation présente ne peut être optimale dans le cas général. Les *états estimés*[10] sont des états d'information couramment utilisés pour définir les politiques car on sait qu'ils permettent de ramener un POMDP à un MDP (Aström, 1965) et donc de trouver des politiques optimales. Mais ces états estimés sont en fait des distributions de probabilités sur l'espace des états $\mathcal{S}$ et posent plusieurs problèmes qui ne sont que partiellement résolus et donc toujours d'actualité.

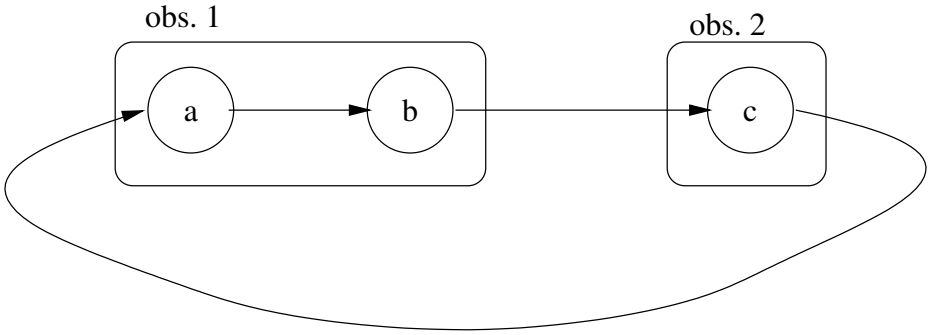


FIGURE 3.2 – **Processus non markovien dans un POMDP.** Dans ce POMDP très simple et déterministe il n'y a que 3 états (les "ronds" a, b et c), deux observations (les "rectangle" A et B) et une seule action. Les états a et b produisent l'observation A et l'état c produit l'observation B. La séquence des observations ne définit pas un processus markovien car la probabilité de transition à partir de l'observation A dépend de l'observation précédente. Plus formellement, au temps t, on a par exemple : $\Pr(o_{t+1} = A | o_t = A) \neq \Pr(o_{t+1} = A | o_t = A, o_{t-1} = A)$.

D'une part, les états estimés nécessitent la connaissance du modèle du POMDP pour être utilisés. En effet, la connaissance de l'état estimé au temps $t+1$ nécessite une inférence bayésienne à partir de l'état estimé au temps t, de l'action qui vient d'être faite et de l'observation courante. Comme le montre l'équation (3.1), cette inférence nécessite la connaissance de la dynamique de l'environnement et notamment des fonctions de transition et d'observation ($p()$ et $\Omega()$). Dans un cadre d'apprentissage, il faudrait soit apprendre le modèle (voir section 3.3) soit apprendre aussi à prédire un état estimé.

$$\begin{aligned} b_o^a(s_{t+1}) &= \Pr(s_{t+1} | b_t, a_t, o_{t+1}) \\ &= \frac{O(o_{t+1}|s_{t+1}) \sum_{s \in \mathcal{S}} p(s_{t+1}|a_t, s) b_t(s)}{\sum_{s \in \mathcal{S}} \sum_{s' \in \mathcal{S}} O(o_{t+1}|s') p(s'|a_t, s) b_t(s)}. \end{aligned} \tag{3.1}$$

D'autre part, les états estimés définissent un ensemble continu et donc infini. Trouver une

10. De l'anglais "belief states".

politique optimale sur cet ensemble est un problème PSPACE-difficile même en horizon fini et avec une distribution initiale réduite à un dirac (Papadimitriou, 1994). Les algorithmes exacts qui existent, citons notamment WITNESS (Cassandra et al., 1994), ITERATIVE PRUNING (Zang and Lio, 1996) et "IMPROVED POLICY ITERATION" (Hansen, 1998), ne sont utilisables en pratique que sur des exemples avec une dizaine d'états et pour un horizon très réduit. Ils ont néanmoins permis l'éclosion d'algorithmes approchés qui sont capables de résoudre des POMDP de grande taille, parfois sur des horizons assez longs (Baxter and Bartlett, 2000; Guestrin et al., 2001; Pineau et al., 2003; Spaan and Vlassis, 2005).

Plus récemment, les représentations par états prédictifs (PSR pour "*Predictive State Representation*", voir (Littman et al., 2002; Singh et al., 2004)) ont amené un nouvel angle d'approche pour représenter de manière compacte un POMDP. Cette représentation, qui est proche d'un modèle proposé par Jaeger (2000), s'appuie sur des *tests* qui sont des séquences d'actions et d'observations dont on connaît la probabilité des occurrences futures à partir de l'instant présent. A l'aide d'un ensemble fini de tests dont la prédiction est connue, on peut calculer la probabilité d'occurrence de n'importe quel test. Une telle représentation compacte est bien évidemment intéressante mais coûteuse à maintenir à jour car, à chaque pas de temps, il faut recalculer les probabilités de tous les tests de l'ensemble représentatif. De plus, il est généralement nécessaire de connaître le modèle pour chercher les PSR et la fonction de projection qui permet leur mise à jour, bien que des travaux récents se soient intéressés à l'apprentissage et à la découverte de PSR (McCracken and Bowling, 2005; Wingate and Singh, 2007).

Les préoccupations précédentes concernent la recherche de représentations *adéquates* pour les POMDP, c'est-à-dire des représentations permettant de définir des politiques optimales tout en restant tractables. Dans le cadre plus général de l'apprentissage par renforcement, des difficultés supplémentaires liées au fait que le modèle n'est pas connu s'ajoutent à ces problèmes de représentation. Nous allons d'abord examiner ces difficultés supplémentaires avant de voir les rares pistes qui ont été explorées pour y remédier, pistes que j'ai moi-même empruntées par le passé.

3.2 Pourquoi c'est compliqué d'apprendre dans les POMDP

Quand on ne dispose pas d'un modèle de l'environnement, la seule solution pour apprendre une politique est d'interagir avec l'environnement. Comme nous l'avons vu précédemment (cf. section 2.3.3), ces interactions peuvent avoir pour finalité d'apprendre le modèle de l'environnement ou d'apprendre directement une politique, par le biais éventuel d'une fonction de valeur.

Le problème principal vient du fait que, tant que l'agent apprenant n'a pas trouvé une représentation adéquate du processus qu'il est en train d'apprendre, ce dernier apparaît toujours comme non-markovien à l'agent. De ce fait, il y a une interdépendance forte entre la politique que l'agent est en train d'utiliser pour explorer l'environnement et l'évolution de ce dernier, du moins du point de vue de l'agent. Dans le cas d'un MDP, l'évolution du processus ne dépend que de la dernière action effectuée par l'agent et le principe de localité de Bellman permet de montrer que les algorithmes d'apprentissage vont converger vers la politique optimale. Pour un POMDP,

il peut être nécessaire de tenir compte de *tout* l'historique de l'interaction quand on construit la réprésentation, ce qui est rapidement irréaliste. Et, dans le cas d'une représentation inadéquate qui resterait non markovienne, rien ne peut assurer que les algorithmes classiques utilisés pour les MDP vont permettre de trouver la solution optimale du POMDP.

Pour illustrer ce phénomène, nous pouvons nous appuyer sur les travaux de Littman (1994b), approfondis plus tard par Singh et al. (1994), qui explorent l'apprentissage de politiques "sans-mémoire" dans les POMDP, c'est-à-dire des politiques définies sur les seules observations. A cause de la dépendance entre le processus et la politique d'exploration, il n'est pas possible de définir une fonction de valeur qui serait globalement maximale pour *toutes* les observations. Dit autrement, si on cherche une politique sans mémoire pour maximiser la valeur d'une observation, il existe des observations pour lesquelles cette politique n'est pas optimale (voir figure 3.3). Dès lors, les algorithmes classiques (comme le Q-Learning), qui dépendent de l'existence d'une fonction de valeur optimale, peuvent ne pas converger. Même si l'on étend la recherche à des politiques stochastiques, les meilleures politiques trouvées seront généralement sous-optimales, que l'on recherche directement des politiques comme dans (Baxter and Bartlett, 2000; Baxter et al., 2001) ou en passant par une fonction valeur (Jaakkola et al., 1994). Le chapitre de livre consacré aux POMDP que nous avons co-écrit avec Bruno Scherrer revient plus en détails sur les difficultés rencontrées par ces méthodes d'apprentissage dites "adaptées" à partir des méthodes sur les MDP, voir (Groupe PDMIA, 2008).

Ces complications dues à l'interdépendance entre processus et politique s'appliquent aussi à l'apprentissage de modèle de POMDP. Ainsi, d'une manière ou d'une autre, quand ils ne cherchent pas à optimiser les paramètres d'un modèle ou d'une politique, les algorithmes d'apprentissage proposés pour les POMDP recherchent implicitement ou explicitement une représentation interne du POMDP qui leur permette de pouvoir limiter les relations de causalité entre l'historique du processus et son évolution. C'est le cas des nombreux algorithmes décrits dans la revue bibliographique quasi-exhaustive écrite par Hasinoff (2002), revue à laquelle il ne manque que quelques travaux, comme par exemple (Kimura et al., 1997), (Lanzi, 2000) ou (Theocharous et al., 2001). C'est aussi le cas dans les travaux plus récents qui proposent des algorithmes pour apprendre les ensembles de tests qui forment des PSR, comme (Singh et al., 2003), (James and Singh, 2004), (McCracken and Bowling, 2005) et (Aberdeen et al., 2007).

Dans ce qui suit, je reviens plus en détails sur certaines approches d'apprentissage auxquelles j'ai contribué. Ainsi, je m'attarderai d'abord sur une approche indirecte cherchant à apprendre le modèle du POMDP. Puis je parlerai de mes travaux pour tenter de définir des conditions suffisantes pour que l'apprentissage directe d'une fonction de valeur optimale soit possible avant de terminer par une approche plus expérimentale mais finalement assez efficace.

3.3 Une approche indirecte

Comme nous l'avons vu précédemment (section 2.3.3), une voie classique en apprentissage par renforcement est l'approche indirecte qui consiste à apprendre le modèle et à calculer , simultanément ou dans un deuxième temps, une politique optimale. La communauté de la reconnaissance de la parole a beaucoup contribué au domaine de l'apprentissage de modèle pour les chaines

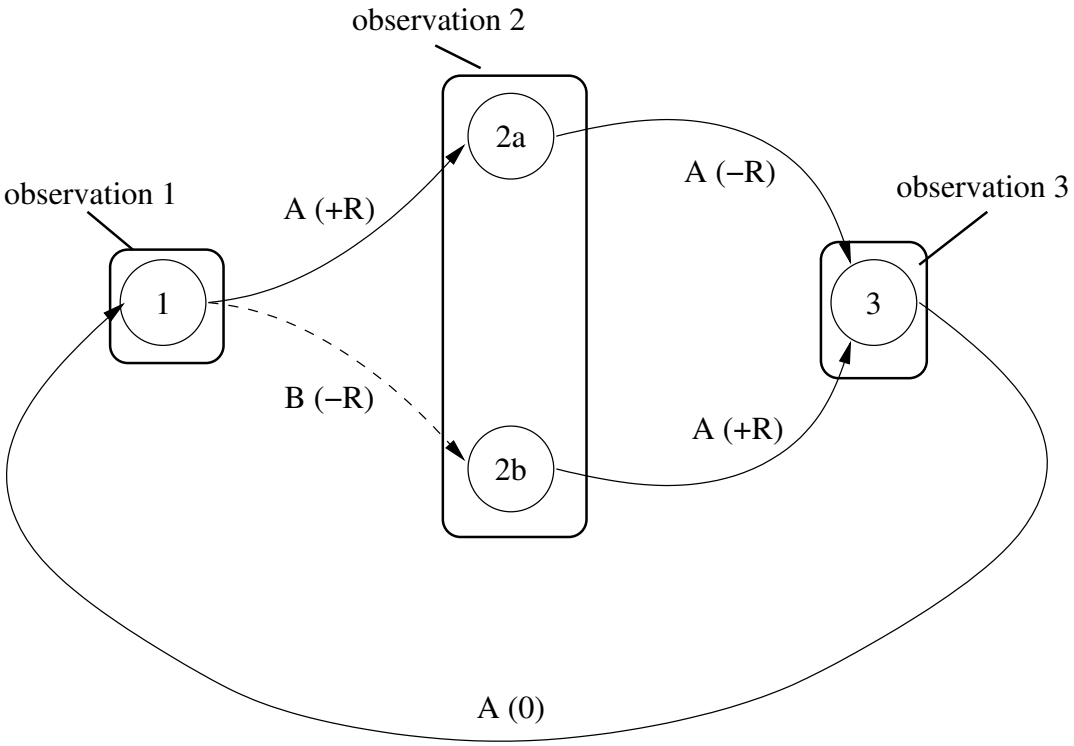


FIGURE 3.3 – **Pas de fonction de valeur sur les observations *globalement* optimale dans un POMDP.** Le POMDP de cette figure est constitué de quatre états (1, 2a, 2b et 3) et de trois observations (1, 2 et 3). Les actions sont symbolisées par des lettres majuscules avec, entre parenthèse, la récompense reçue par l'agent pour le choix de cette action. La seule décision concernant la politique est faite pour l'observation " 1 ". Y augmenter la probabilité de choisir l'action " A " augmente la valeur de l'observation " 1 " et diminue celle de l'observation " 2 ". C'est un exercice assez facile de montrer qu'on ne peut donc trouver de politique maximisant la valeur de *toutes* les observations. (Tiré de (Groupe PDMIA, 2008, Chap. 3)).

de Markov cachées (HMM pour "Hidden Markov Models") qui sont largement utilisées dans cette communauté. Les méthodes s'appuient sur les algorithmes issus des travaux de Rabiner and Juang (1986), qui sont des variantes des algorithmes EM ("Expectation-Maximization"). En ajoutant la notion d'action aux HMM, on peut étendre l'algorithme de Baum-Welch des HMM aux POMDP pour apprendre les probabilités de transition et d'observation dans un POMDP, ce qu'a d'ailleurs proposé Chrisman (1992).

Les choses ne sont pas aussi simples et le problème est en fait loin d'être résolu. Apprendre un modèle peut se décomposer en apprendre la structure du modèle et apprendre les paramètres pour une structure donnée. Le problème de l'apprentissage de la structure est le plus délicat et pourtant de première importance. En fait, sans un *a priori* fort sur la structure du modèle, l'algorithme de Baum-Welch n'est pas très performant. Il requiert un très grand nombre de données, converge lentement et est souvent piégé dans des maxima locaux. Dans le cas de la reconnaissance de la parole, le fait d'utiliser des chaînes de Markov linéaires simplifie grandement

l'apprentissage du modèle, d'autant plus que les modèles comprennent assez peu d'états. Dans le cas des POMDP, où on ne sait même pas à l'avance le nombre d'états sous-jacents, l'utilisation des mêmes méthodes est impossible en pratique.

Il reste possible de travailler avec des *a priori* plus ou moins complexes sur la structure du POMDP. Le plus simple est d'imposer la structure du POMDP comme le font Hoey and Little (2004) mais, même dans ce cas, la complexité des algorithmes de planification est telle qu'ils se contentent ensuite d'utiliser une méthode de *Q-Learning adapté*[11] pour trouver une politique acceptable. Certains ont aussi travaillé avec des modèles déterministes (Amir, 2005). Dans un cadre particulier, il est possible d'exploiter la structure particulière du problème dans des *a priori* plus ciblés, comme par exemple en robotique mobile où la géométrie de l'environnement et les mesures odométriques permettent de mieux cerner la structure du modèle (Shatkay, 1999). Cette idée de disposer d'informations "extérieures", en une sorte "d'apprentissage actif", est analysée plus formellement en utilisant un oracle qu'il est possible d'interroger pendant l'apprentissage (Jaulmes et al., 2005).

Dans un cadre bayésien, j'ai réfléchi à partir d'une idée originale exposée par Stolcke and Omohundro (1992, 1994) pour les HMM. Leur algorithme construit d'abord un modèle naïf où chaque nouvelle observation est considérée comme liée à un nouvel état, ce modèle est ensuite restructuré en essayant de fusionner les états jugés similaires et en s'appuyant sur un *a priori* concernant la structure du modèle final. Il me semble que leur approche pourrait être étendue aux POMDP. Un travail intéressant serait alors de porter cet algorithme au cas des POMDP, de tester méthodiquement cette approche pour déterminer, en particulier, son efficacité et appréhender sa dépendance au choix des différents *a priori*.

Ces considérations sur l'apprentissage de la structure d'un modèle sont encore au cœur des problématiques de recherche actuelles, ainsi que le montre l'activité de la communauté des modèles graphiques sur le sujet (Jordan, 1999). En effet, les modèles graphiques généralisent les processus décisionnels observables ou non.

3.4 Une approche directe

Lors de ma thèse (Dutech, 1999), j'ai abordé le problème de l'apprentissage par renforcement dans les POMDP de manière directe. Je suis parti du constat que les états estimés sont une sorte de condensé du passé du processus et contiennent toutes les informations passées "pertinentes" qui permettent de décider de l'action optimale. Une idée "naturelle" est alors de doter l'agent apprenant d'une mémoire de ses actions et observations passées, c'est-à-dire de chercher des politiques définies sur l'espace des trajectoires d'actions-observations. Une façon naïve d'éviter l'explosion combinatoire liée avec la croissance exponentielle du nombre d'états ainsi étendus par des historiques est de limiter arbitrairement la taille de ces historiques. J'ai mis au point un algorithme plus fin qui tire parti du fait que, dans certains cas, l'information pertinente est en fait assez récente : la taille de l'historique nécessaire pour décider de l'action optimale dépend de l'état sous-jacent du processus. La figure 3.4 illustre cette notion.

11. utilisation du *Q-Learning* en faisant comme si les observations étaient des états

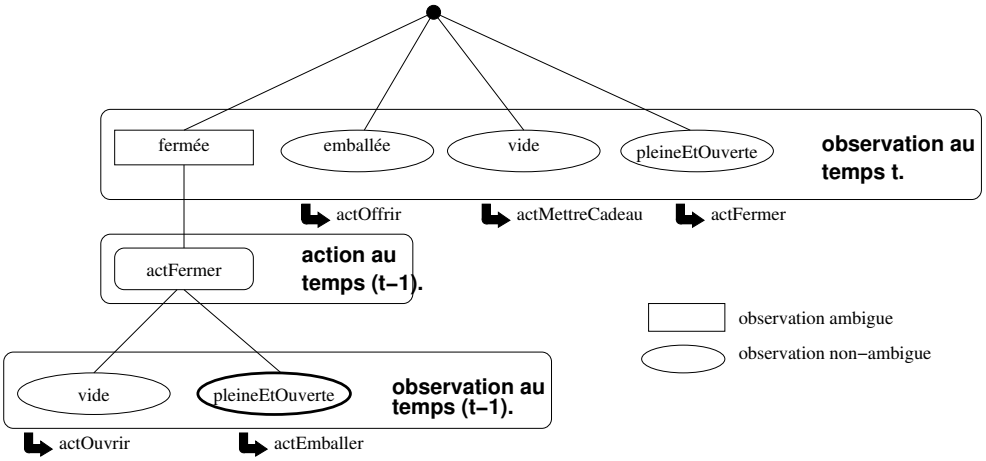


FIGURE 3.4 – **Historiques pertinents pour POMDP.** Cet arbre représente une politique définie sur des trajectoires d'actions-observations pour un problème où un agent doit apprendre à emballer un cadeau. Les observations de l'agent qui regarde la boite à cadeau sont *fermée, emballée, OuverteEtVide, OuverteEtPleine*. Si la plupart de ces observations sont suffisantes pour agir optimalement, dans le cas où la boite est *fermée*, il faut se "souvenir" du passé du processus pour savoir comment agir. On voit que selon l'observation, la taille de l'historique nécessaire pour agir optimalement n'est pas la même. (Tiré de (Dutech and Samuelides, 2003))

J'ai d'abord formalisé et prouvé le bien fondé de cette approche en développant le concept d'*observable exhaustif*. Si on a trouvé un observable exhaustif pour un POMDP, j'ai montré qu'il est possible de définir une politique optimale à partir des historiques maximaux dérivés de cet observable. Le principe de l'algorithme d'apprentissage par renforcement que j'ai développé est de construire de manière incrémentale les historiques "pertinents" formant un observable exhaustif (ils ne sont pas tous de même taille au sein d'un observable) et, *en même temps*, d'estimer la fonction valeur des couples (historique, action). La principale difficulté de l'algorithme est de détecter les historiques *ambigus* afin de les étendre, ce qui est fait de manière heuristique. Cet algorithme d'apprentissage "en-ligne" a été validé sur des exemples de complexité croissante. L'approche que j'ai suivie est très proche de celle détaillée dans (McCallum, 1996). Il y utilise des heuristiques plus sophistiquées mais ne s'assure pas du fondement théorique de sa méthode et ne peut donc garantir que son algorithme permet de trouver des solutions.

Quand je suis arrivé au LORIA, j'ai étendu cette méthode au cas de la planification dans les POMDP, c'est-à-dire quand le modèle du POMDP est connu. Plutôt que d'apprendre un observable exhaustif, il paraît intéressant de le *calculer* pour en déduire une extension d'état permettant de mettre en œuvre des algorithmes de planification ou d'apprentissage classiques. C'est à cette occasion que nous avons montré que, dans le cas général, il n'existe pas d'observable exhaustif car on peut toujours trouver des historiques pertinents qui sont de longueur infinie. Il faut par exemple que l'agent s'impose des restrictions, parfois naturelles ou intuitives (comme ne pas alterner indéfiniment entre deux actions), pour qu'existe alors un observable exhaustif. Ces travaux et ces constatations sont développées dans un article publié à ECAI (Dutech, 2000).

La notion d'observable exhaustif est importante car elle offre une formalisation d'un principe souvent utilisé qui consiste à mémoriser les événements du passé pour mieux prédire le présent. De plus, pour de nombreux processus, même s'il n'existe pas d'observable exhaustif, il existe une politique optimale qui est définie sur un observable *fini*. Ainsi, il est important de noter que, lorsque l'agent essaye d'apprendre *en ligne* un observable exhaustif, l'aspect "ancré dans la réalité" de l'algorithme d'apprentissage lui permet d'éviter d'essayer d'apprendre les historiques infinis et augmente sa probabilité de trouver quand même un comportement optimal. C'est une illustration un peu particulière du fait que c'est bien par ses interactions répétées avec son environnement qu'un agent peut apprendre alors que, dans le cas présent, un raisonnement "hors ligne" (déconnecté du monde) s'avère moins efficace voire impossible.

Il découle de la dernière affirmation une possibilité immédiate de perspective à mes travaux. Plutôt que de vouloir calculer un observable exhaustif, une approche plus raisonnable serait de procéder de manière incrémentale en alternant des phases de calcul d'observables et des phases d'estimation de la qualité des politiques déduites de ses observables. Un critère d'arrêt heuristique est tout de même indispensable. Une autre extension est liée à la limitation majeure de l'approche suivie due à la taille croissante de l'espace d'états étendus constitué par les historiques "pertinents". Lors d'un stage d'initiation à la recherche avec quelques étudiants, nous avons travaillé sur le concept de *contexte pertinent* dans le but d'étendre l'état non plus par des historiques complets (une mémoire de *tous* les évenements de t à $t-k$) mais par des listes de certaines observations passées jugées pertinentes. Le problème de décider de la pertinence d'une observation passée est en fait extrêmement difficile et n'a pas été résolu, comme le montre Shalizi dans sa thèse (Shalizi, 2001). On peut considérer que c'est en fait un des problèmes *fondamentaux* en apprentissage, voire en intelligence artificielle. Néanmoins, bien que ces travaux aient été peu concluant, Joseph Razik, l'un des étudiants, a pris goût à la recherche, a tel point qu'il a soutenu une thèse dans l'équipe Parole du Loria en 2007.

3.5 Une approche "constructiviste" ou "développementale"

Avec Olivier Buffet, nous avons voulu nous intéresser davantage à la construction par l'agent de représentations adéquates pour lui. Au cours de sa thèse, que j'ai co-encadrée et qu'il a soutenue en 2003 (Buffet, 2003), nous avons abordé ce problème sous un angle que l'on pourrait qualifier de développemental [12], ce qui était alors assez original dans le domaine de l'apprentissage par renforcement.

Nous sommes partis de l'idée que le comportement d'un agent, surtout quand il doit effectuer une tâche complexe, peut être le résultat de la combinaison de plusieurs sous-comportements plus simples. Par exemple, pour pousser une tuile en évitant des obstacles, le comportement d'un agent pourrait résulter de la combinaison des deux sous-comportements `pousser-tuile` et `eviter-trou`. Pour plus de flexibilité, le comportement global ne se réduit pas à une alternance entre l'un ou l'autre des sous-comportements. Nous avons proposé une méthode où le comportement global tient compte simultanément des sous-comportements en pondérant les fonctions valeurs de chacun de ces sous-comportements pour obtenir la fonction valeur globale d'un état.

12. en référence à la robotique "dévelopementale" qui est en plein développement (Lungarella et al., 2003)

Les poids valuant chacun des sous-comportements sont appris par renforcement, ce qui est un problème difficile en soi (voir (Buffet et al., 2002b)).

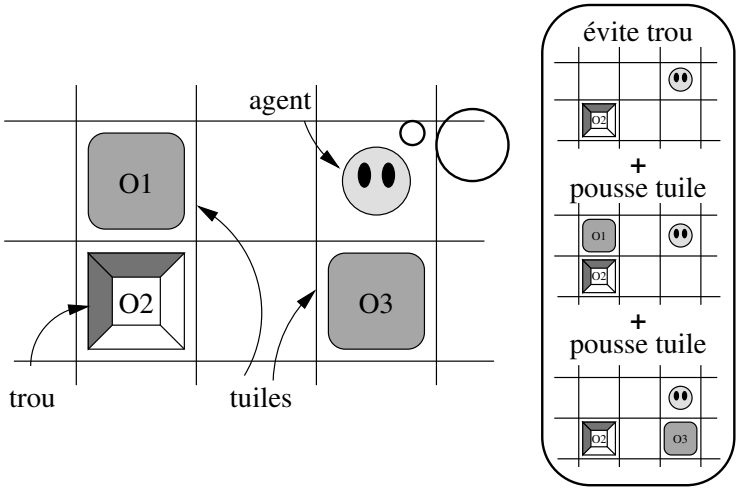


FIGURE 3.5 – **Combinaison de sous-comportements**. L'agent doit apprendre à pousse les tuiles dans les trous. Pour cela, il va apprendre à combiner deux type de comportements de base : `éviter` *`un`* `trou` et `pousser` *`une`* `tuile dans` *`un`* `trou`. Le deuxième comportement sera instancié deux fois, une fois avec la tuile "O1" et le trou "O2", et une fois avec la tuile "O3" et le trou "O2". (Tiré de (Buffet et al., 2006)).

Les sous-comportements sont eux-mêmes appris par renforcement d'une manière assez originale puisque l'agent définit lui-même les états de ces sous-comportements en apprenant en même temps une représentation pertinente de son environnement (par exemple, "il faut tenir compte de 2 tuiles", "il faut tenir compte d'un trou", *etc*). De plus, une combinaison particulièrement utile de sous-comportements peut ensuite être considérée comme un nouveau sous-comportement en tant que tel. Les sous-comportements ainsi appris sont de plus en plus complexes, soit parce qu'ils sont associés à une représentation faisant appel à un nombre croissant de caractéristiques de l'environnement, soit parce qu'ils combinent eux-mêmes plusieurs sous-comportements.

Ainsi, en interagissant avec l'environnement, l'agent se construit une représentation subjective pertinente de son environnement et un comportement lui permettant de résoudre des tâches de plus en plus complexes. C'est en cela que la démarche peut être qualifiée de développementale. La méthode pour apprendre de nouveaux sous-comportements est restée assez simple car, dans les faits, elle s'apparente presque à une recherche exhaustive dans l'espace des sous-comportements (Buffet et al., 2004).

Comme nous l'avons argumenté dans (Buffet et al., 2004), le cadre posé reste intéressant pour plusieurs points.

- il est générique car l'agent est censé être capable de se contruire, à partir de perceptions très basiques, des comportements et des représentations guidées par la tâche ;
- il facilite le transfert de connaissances entre différentes tâches puisque l'agent peut réutiliser des sous-comportements en les valuant différemment ;
- il est adaptatif car l'agent peut, en fonction de ses besoins, changer la valuation des sous-comportements ou apprendre de nouveaux sous-comportements ;

- il passe assez bien à l'échelle car les représentations construites par l'agent sont subjectives, l'agent n'est pas censé connaître l'état de tout l'environnement ;
- il permet à un agent d'apprendre des comportements utiles et pertinents bien que ce dernier soit confronté à un POMDP.

Le travail que nous avons réalisé pourrait être étendu en s'inspirant ou en le combinant avec d'autres travaux plus récents. Dans le domaine de la "sélection d'action" dont nous nous sommes déjà largement inspirés, des travaux récents suggérent de moduler l'importance des percepts en fonctions de *plusieurs* motivations pour adapter le comportement global (Avila-Garcia and Canamero, 2004). Un des buts est d'améliorer le comportement global qui tombe souvent sur des oscillations dues à des compromis entre des actions opportunistes et des comportements plus persistants.

Les travaux proposant d'apprendre des contrôleurs hiérarchiques pour les MDP et les POMDP, ce qui est très proche de notre cadre, s'intéressent souvent à montrer que l'approche est toujours optimale. Dans certains cas l'accent a été mis sur la découverte de sous-buts ou de sous-contrôleurs (comme par exemple le HQ-Learning de Wiering and Schmidhuber (1996) ou les travaux de (Digney, 1998), (Dietterich, 2000)). Là encore, des travaux récents argumentant que le critère de performance moyen serait plus adéquat que le critère pondéré en terme de performance et de détection de sous-problèmes seraient à regarder de plus près (Ghavamzadeh and Mahadevan, 2007). Cette constatation est à mettre en relation avec les travaux théoriques de Singh et al. (1994) qui montrent qu'un critère moyen est le seul théoriquement utilisable dans le cas de politiques sans-mémoire.

3.6 Où en sommes-nous maintenant ?

De mes travaux et des autres travaux sur le sujet, ce qui ressort de l'apprentissage par renforcment pour les POMDP est que beaucoup de choses reposent sur la qualité de la représentation que l'agent se fait de son environnement. Si cette représentation est trop pauvre, par exemple quand l'agent ne considère que les observations présentes du processus, il est généralement impossible d'apprendre une politique optimale. Si cette représentation est trop riche, elle sera rapidement trop lourde à gérer et, pour chaque situation, il y aura trop peu d'exemples pour que l'agent puisse apprendre.

Les difficultés essentielles à surmonter pour construire cette représentation sont liées à la recherche de régularités dans la dynamique complexe formée par l'agent et son environnement. Les outils de l'apprentissage par renforcement s'appuient fortement sur la théorie de l'approximation stochastique pour détecter et "modéliser" ces régularités. On peut légitimement se poser la question de savoir si des méthodes statistiques sont suffisantes pour détecter les "bonnes" régularités. Peut-on se passer de la notion de "sens" ?

La notion de "sens" telle que je l'entends, et tel que l'*embodiment* l'entend, ne passe pas par le langage ou le symbole. De manière moins abstraite mais tout aussi ambitieuse, il me semble plus réaliste de s'attacher à ce que l'agent se construise des représentations adaptées à ses interactions avec l'environnement, c'est-à-dire des boucles sensori-motrices lui permettant de mener à bien ses tâches. La piste que nous avons commencé à suivre et qui consiste à adapter les interactions

aux représentations apprises par l'agent est un exemple de ce genre de démarche.

Il est important de respecter la contrainte d'autonomie des agents. Dans ce cadre, comment faciliter l'apprentissage de représentations pertinentes et adéquates ?

Une voie de recherche possible est de s'intéresser à la motivation intrinsèque d'un agent pour apprendre et diriger son apprentissage. Dans le cadre de l'apprentissage, cela se traduit par une génération, par l'agent lui-même, d'un signal de récompense. Ce signal est une sorte de signal de "méta-apprentissage" '. Les questions ouvertes sont des plus nombreuses, par exemple :

- Est-ce un signal "inné" ou "acquis" par l'agent ?
- Est-ce que ce signal peut lui-même évoluer en fonction des comportements appris par l'agent ?
- Est-ce que l'apprentissage par renforcement, comme méthode d'apprentissage et de méta-apprentissage est *suffisant*. Qu'y ajouter ?

Je veux également mentionner le domaine de recherche de la robotique développementale, discipline lancée en 2000 par un article collectif (Weng et al., 2000) et en plein essor. On s'y intéresse notamment à "l'auto-acquisition" par un agent de comportements par le biais de motivations internes. Les travaux de Oudeyer et al. (2007) s'inscrivent dans le cadre de l'apprentissage par renforcement et donnent des pistes très intéressantes pour que l'apprentissage d'un agent soit à la fois guidé et motivé par la découverte de comportements à sa portée, c'est-à-dire ni trop simples, ni trop compliqués.

Une autre voie passe par des interactions plus ou moins dirigées avec d'autres agents, humains ou non. Les autres agents peuvent avoir le rôle de professeur, d'expert et l'agent apprenant doit avoir une certaine motivation à imiter, reproduire ou s'inspirer des autres agents. Il est aussi concevable que tous les agents soient en train d'apprendre et c'est par certains mécanismes de transfert et d'échanges que l'ensemble pourra apprendre.

C'est dans l'optique de mieux comprendre les mécanismes mis en jeu dans les phénomènes d'apprentissage collectif que je me suis intéressé à l'utilisation de l'apprentissage par renforcement pour les systèmes multi-agents, ainsi que nous allons le voir maintenant.

4

Des agents qui apprennent ensemble

Plusieurs motivations m'ont amené à m'intéresser à l'apprentissage dans le cadre des systèmes multi-agents. Il y a d'abord le fait que, comme nous le verrons, un agent au sein d'un système est confronté à un problème non-markovien. De plus, il est indéniable qu'une bonne partie du développement de l'intelligence, du moins en ce qui concerne l'homme, est rendue possible par le fait qu'il est entouré d'autres agents humains. C'est bien parce que nous interagissons avec d'autres que nous acquérons de nombreuses capacités cognitives et en particulier le langage. C'est pourquoi, bien que cette approche risque de ne pouvoir porter ses fruits qu'à très long terme, il me semble important de m'intéresser à l'apprentissage dans les systèmes multi-agents sous l'angle de l'*embodiment* et dans la perspective de la compréhension des mécanismes de la cognition.

Parmi les nombreux types de systèmes multi-agents, mes travaux concernent uniquement des systèmes constitués d'agents simples où chaque agent apprend ses comportements par apprentissage par renforcement. Ainsi, je n'ai pas abordé – pour l'instant – l'intelligence en essaim, aussi appelé "*swarm intelligence*" (Bonabeau et al., 1999) car les agents dans ces types de systèmes n'ont *a priori* pas les capacités cognitives suffisantes pour utiliser — en ligne — les méthodes classiques d'apprentissage. Et je ne m'intéresse pas non plus aux systèmes composés d'agents (trop) cognitifs, comme ceux qui sont visés dans (Weiss and Sen, 1996), car les agents y sont cette fois plus complexes que nécessaire, du moins en ce qui concerne mes objectifs.

Les agents que nous considérons peuvent être plus finement caractérisés et les propriétés suivantes sont importantes dans le cadre de mes travaux.

- **Les agents sont situés**. Les agents ont une perception locale et limitée de leur environnement et des autres agents. Ils ne peuvent que se forger, au mieux, une représentation subjective de leur monde.
- **Les agents sont indépendants**. Il n'y pas ou très peu de communication entre eux et chaque agent apprend, décide ou planifie par lui-même, indépendamment des autres ou d'une aide extérieure.
- **Les agents sont autonomes**. L'agent ne dispose pas d'une aide extérieure ou humaine pour apprendre, décider ou planifier.
- **Les agents sont coopératifs**. Les agents, bien que n'en ayant une vision au mieux parcellaire, essayent tous de contribuer à la réalisation d'une tâche globale pour laquelle il

faut en général de la collaboration et de la coordination. Ces tâches globales ne peuvent être résolues par un agent seul.

Le point sur lequel je veux insister, outre le fait que chaque agent se trouve en général devant une tâche non-markovienne à résoudre, est l'existence de deux niveaux de description et d'abstraction dans le système. Au niveau individuel, on trouve les agents qui n'ont qu'une vue très partielle des autres et de leur environnement ; ils peuvent même ne pas connaître la tâche globale à résoudre. Au niveau global, il y a une tâche à résoudre mais aucun agent n'a une représentation ou n'opère à ce niveau. C'est vraiment des interactions entre les agents que la réalisation de la tâche globale va émerger.

Nous allons d'abord examiner quelles sont les problématiques supplémentaires posées par ce cadre de travail avant d'explorer les travaux que nous avons développés pour tenter d'apporter des réponses à ces problématiques. Ainsi, nous tenterons de mieux comprendre les mécanismes à mettre en jeu par le biais de la planification dans le formalisme mathématique des POMDP décentralisés. Nous ferons un détour par la théorie des jeux pour voir en quoi elle peut nous aider à trouver des réponses avant d'aborder une théorie incrémentale de l'apprentissage qui me paraît prometteuse. Le tout se concluera par un petit bilan sur cette partie.

4.1 Difficultés spécifiques au cadre multi-agent

Le cadre formel des MDP permet théoriquement de modéliser un système, même quand il est composé de plusieurs agents. L'état est alors le produit de l'état de chacun des agents et de l'environnement et une action est composée de l'ensemble des actions des agents. On parle alors d'action jointe. C'est exactement le cadre de l'étude de Boutilier (1996) sur ce qu'il appelle les MMDP ("Multi-agent Markov Decision Process") et il y relève deux difficultés. La première difficulté concerne la taille des espaces d'états et d'actions qui augmente de manière exponentielle avec le nombre d'agents, ce qui limite sérieusement le passage à l'échelle des algorithmes de résolution, comme nous l'avons montré dans la section 2.4.1.

La deuxième difficulté est plus cruciale et s'intéresse aux agents qui planifient — étant dans le cadre des MMDP, les agents ont tous une connaissance de la dynamique du monde — chacun indépendamment des autres. Chaque agent, qui connaît l'état complet et les actions possibles de tous les agents, dispose d'une connaissance totale du système et pourtant le plan collectif calculé par un agent peut être différent du plan collectif trouvé par un autre agent car un MDP peut avoir plusieurs politiques optimales de même valeur.

Par exemple, imaginons deux agents devant se croiser dans un couloir. Pour ne pas se heurter, ils doivent soit tous les deux faire un écart sur leur gauche, soit tous les deux un écart sur leur droite. Ce sont là deux politiques optimale de même fonction de valeur. Chacun des deux agents peut très bien penser qu'il agit optimalement et, lors de l'exécution de ces politiques jointes, les agents vont pourtant entrer en collision (voir figure 4.1). Ainsi que l'expose Boutilier, des agents *indépendants* doivent disposer d'un moyen pour se *coordonner* et c'est un problème très difficile pour des agents autonomes et indépendants sans *a priori* sur leur environnement. En effet, pour se coordonner les agents peuvent :

– faire appel à coordinateur central. Cette option ne rentre pas dans le cadre des SMA et pose

la question de comment le coordinateur central a développé sa capacités à coordonner ;
- disposer de conventions établies *à l'avance* comme, par exemple, la convention de rouler à droite sur les routes. C'est peu compatible avec des agents autonomes et encore moins avec l'*embodiment* qui m'intéresse car se pose alors la question de savoir comment les agents ont acquis ce savoir (on retrouve le problème de l'homoncule présenté en section 2.1) ;
- communiquer entre eux pour choisir une politique jointe unique et commune. Il faudrait donc que les agents soient déjà dotés de capacités de communication, donc qu'ils disposent d'un langage commun. On se retrouve encore avec le problème de savoir comment ce langage a été acquis, ce qui est en fait une des facettes de la problématique générale à laquelle je m'attache ;
- apprendre à se coordonner. Le problème d'apprentissage est déplacé mais reste entier. Pour reprendre le problème du couloir de la figure 4.1, si les agents ont appris les différentes comportements possibles il reste néanmoins à apprendre comment choisir tous les deux le même comportement optimal. Les agents sont toujours faces aux réelles difficultés de l'apprentissage multi-agent : le problème est non-markovien et non-stationnaire, comme nous allons le détailler ci-dessous.

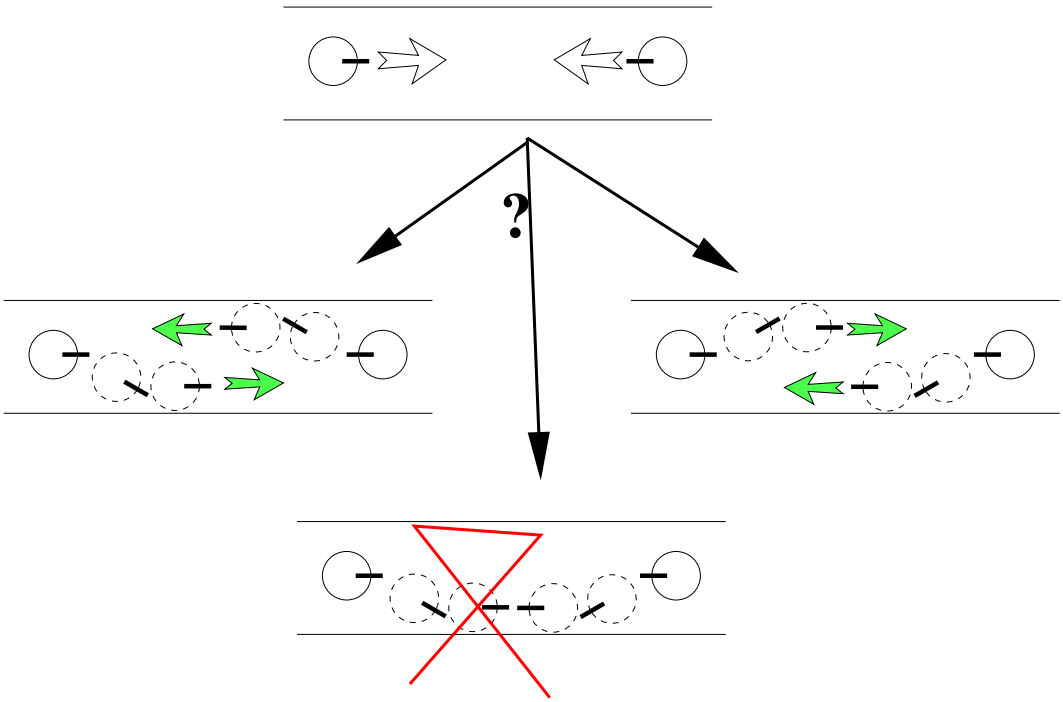


FIGURE 4.1 – **Croisement dans un couloir**. Deux agents doivent se croiser dans un couloir. Ils sont indépendants et ne peuvent communiquer. Deux "plans" optimaux s'offrent à eux. Soit ils se poussent tous les deux sur leur gauche, soit tous les deux sur leur droite. Sans convention préétablie, sans communication et sans décideur global, comment vont-ils faire pour éviter la solution blocante que l'on retrouve au bas de la figure ?

Dans les systèmes multi-agents qui m'intéressent, les agents n'ont qu'une vue locale et donc généralement partielle de leur monde. Ils ne perçoivent que leur environnement immédiat et les agents qui sont proches. Dès lors, du point de vue d'un agent, le monde perçu est généralement

non-markovien et je serais tenté de dire que les "causes" empêchant le problème d'être markovien sont encore plus nombreuses que dans le cas mono-agent.

A cette difficulté s'ajoute le fait que des agents peuvent être en train d'apprendre. Le comportement de chaque agent peut ainsi être amené à évoluer au cours du temps. Même si l'environnement lui-même est stationnaire, la dynamique du système n'est plus stationnaire, sauf à considérer les comportements de chacun des agents comme faisant partie de l'état du système. En tout cas, du point de vue d'un agent, le problème a une dynamique non-stationnaire.

D'autres extensions des MDP prennent mieux en compte ces spécificités des SMA, notamment en ce qui concerne des agents décentralisés avec des perceptions limitées. Il s'agit par exemple de la famille des modèles décentralisés comme les Dec-MDP ou les Dec-POMDP initialement proposés par Bernstein et al. (2000). Ils ont montré que planifier un comportement optimal pour cette classe de problèmes où les agents cherchent à se coordonner était NEXP-dur, même avec seulement deux agents qui connaissent tout le modèle (probabilités de transition, d'observation, de récompense). Les travaux de thèse de Raghav Aras que j'ai co-encadrés s'intéressent à cette problématique et j'en reparlerai plus en détail en section 4.2.

Les similarités entre les modèles précédents et le cadre de la théorie des jeux (voir, par exemple, (Myerson, 1991; Fudenberg and Tirole, 1991)) invitent à puiser dans la vaste littérature de ce domaine pour mieux analyser les problèmes multi-agents et, éventuellement, proposer de nouvelles méthodes de résolution. La théorie des jeux s'intéresse à des joueurs rationnels engagés dans des jeux où le gain d'un joueur dépend non seulement de sa stratégie mais aussi de la stratégie des autres joueurs. Suivant la forme du jeu, les joueurs sont en concurrence ou alors gagnent à coopérer et à se coordonner. Les travaux concernent essentiellement l'existence et la forme de stratégies collectives, qui sont appelées des équilibres, pour des joueurs rationnels engagés dans un jeu.

De nombreux travaux sur l'apprentissage par renforcement ont cherché à exploiter les concepts définis en théorie des jeux. Les différents types de jeux sont autant de catégories de problèmes d'apprentissage et offrent une taxonomie intéressante pour définir et évaluer les propriétés des algorithmes. Les équilibres offrent aux agents un moyen de se stabiliser sur un comportement collectif. La notion de regret est une autre façon d'évaluer la qualité du comportement d'un agent. Tous ces concepts ont permis de formaliser des notions utilisées informellement ou ont amené de nouvelles façons d'appréhender les problèmes. Nous y reviendrons plus longuement dans la section 4.3 qui détaille, entre autres, les travaux que Raghav Aras et moi-même avons effectué dans cette optique.

Cependant, relativement peu de travaux en théorie des jeux portent sur l'apprentissage lui-même. Une des causes possibles est un résultat récent de Hart and Mas-Colell (2003) qui montre qu'il n'existe pas de méthode générale d'apprentissage pour des agents indépendants. Autrement dit, quel que soit l'algorithme utilisé, il existe des problèmes pour lesquels des agents qui ne sont pas couplés par des communications, des conventions sociales ou d'autres moyens, ne peuvent pas apprendre une politique jointe optimale.

Dès lors, dans le cas d'un problème multi-agent, une solution pour apprendre est d'utiliser les connaissances sur la nature du problème pour doter les agents de capacités et de connaissances leur permettant d'apprendre à résoudre le problème. La difficulté réside dans les choix sur la

quantité d'information à incorporer dans les agents ou dans l'algorithme d'apprentissage. Si cette connaissance, au sens large, fournie aux agents n'est pas suffisante, ils ne pourront pas apprendre une solution au problème. Si trop de connaissance est fournie, il n'y a plus besoin d'apprentissage et le concepteur a fourni suffisamment d'effort pour que le problème posé soit résolu. Il y a là un compromis à trouver, l'idéal étant de disposer de l'algorithme le plus générique possible.

Cette démarche, qui s'apparente à trouver une solution *ad hoc* pour chaque problème, n'est pas très satisfaisante et n'est pas compatible avec ma démarche. Je rappelle ici que je vise, sur le long terme, des agents autonomes, c'est-à-dire des agents auxquels le concepteur ne fournit pas d'information, de connaissance, de données. C'est par lui-même et par ses interactions que l'agent doit trouver des comportements que l'on pourrait qualifier d'intelligents. La seule autre voie possible semble être de se tourner vers des agents qui apprennent à trouver les bons couplages entre eux, de manière à être capable de résoudre tout problème multi-agent posé. C'est une sorte de pari, car l'article de Hart ne démontre nullement que si les agents sont couplés il existe des algorithmes universels. Cependant, c'est une voie pleine de potentiel à explorer et à analyser, ce que j'ai fait de diverses façons qui sont détaillées en sections 4.3 et 4.4.

Une solution que je n'ai pas explorée concerne la capacité pour un agent à apprendre un modèle du comportement des autres agents. C'est un sujet de recherche actuel et difficile, qui touche aux problématiques fondamentales de l'apprentissage dans les systèmes multi-agents : le sujet de l'apprentissage, à savoir le comportement d'un autre agent, évolue constamment et n'est pas totalement perceptible par un agent. On se trouve de nouveau devant un problème non-markovien et non-stationnaire. Ce phénomène de co-évolution explique en partie, comme l'a montré Hu and Wellman (1998b), qu'un agent qui s'appuie sur un modèle imparfait des autres agents de son environnement peut apprendre un comportement qui lui parait "optimal" alors qu'il est pire que celui qu'il aurait pu apprendre s'il n'avait pas disposé d'un modèle des autres.

Il me semble généralement impossible d'obtenir un modèle "parfait" des autres agents puisque, si les autres agents apprennent eux aussi un modèle, on se retrouve devant un paradoxe. Le modèle de l'agent `A` doit tenir compte du comportement de l'agent `B`. L'agent `B` doit tenir compte dans son modèle du fait que l'agent `A` à un modèle de `B`. Et ainsi de suite... Ce processus peut être sans fin et sans solution. Les travaux de Iadine Chadès au sein de l'équipe Maia ont permis de constater que, dans certains cas, malgré une co-évolution orchestrée et centralisée (un seul agent apprend à un moment donné), les comportements appris par les agents ne se stabilisent pas mais changent, souvent de manière cyclique (Chadès, 2003, 2006). Une modification du comportement d'un agent se répercute dans les autres agents jusqu'à modifier le comportement de l'agent, et ainsi de suite.

Mais la raison principale qui fait que je ne me suis guère intéressé à cette problématique particulière est qu'elle suppose des agents dotés de capacités cognitives d'assez haut niveau pour, entre autres, reconnaître et identifier les autres agents, percevoir et identifier les différentes actions des autres agents, *etc.* Encore une fois, je m'intéresse plus aux mécanismes d'apprentissage et d'adaptation qui pourraient, à terme, permettre aux agents d'acquérir des capacités cognitives de plus haut niveau qu'à l'utilisation de telles capacités dans un contexte d'apprentissage.

Et c'est avec cette idée de développement autonome d'un agent au sein d'un système composé de plusieurs agents, et donc confronté à un problème non-markovien et non-stationnaire, que j'ai

exploré les pistes que je détaille maintenant.

4.2 Sans apprentissage, des problèmes déjà complexes

La partie principale de la thèse de Raghav Aras, que j'ai co-encadrée, porte sur une extension très générale des MDP aux systèmes multi-agents, à savoir les Processus Décisionnel de Markov Partiellement Observable Décentralisés (Dec-POMDP) (Bernstein et al., 2000). Une présentation plus détaillée des Dec-POMDP et de divers modèles dérivés se trouve dans (Groupe PDMIA, 2008). Ces modèles considèrent des agents dont les perceptions limitées ne leur permettent pas de connaître l'état global du système, même si les agents mettaient toutes leurs perceptions en commun. Le but de chaque agent est de trouver un comportement *individuel* tel que le comportement *global* du système soit optimal. La qualité de la politique jointe formée par l'agrégation des politiques individuelles des agents est mesurée selon un critère qui dépend d'une fonction récompense définie globalement. Les Dec-POMDP modélisent donc des problèmes coopératifs car les agents ont un but commun.

Formellement, inspiré par les POMDP (voire section 3.1), ce modèle est décrit par la donnée du tuple $\langle \mathcal{S}, N, \mathcal{A}, \Omega, p, O, r, b_0 \rangle$ où

- $\mathcal{S}$ est un ensemble d'états fini ;
- N est un ensemble d'agents, de taille n ;
- $\mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_n$ est un ensemble d'actions fini, composé des ensembles d'actions $\mathcal{A}_i$ de chaque agent i de N ;
- $\Omega = \Omega_1 \times \cdots \times \Omega_n$ est un ensemble d'observations fini, composé des ensembles d'observations Ω_i de chaque agent i de N ;
- $O_i() \ : \mathcal{S} \longrightarrow \Delta(\Omega_i)$ est la fonction d'observation de l'agent i de N. Ainsi, $O_i(o_i|s)$ est la probabilité d'observer l'observation o_i dans l'état global s ;
- $p() \ : \mathcal{S} \times \mathcal{A} \longrightarrow \Delta(\mathcal{S})$ est une fonction de transition. Elle dépend de l'action jointe de tous les agents. Ainsi, $p(s'|(a_1, \cdots, a_n), s)$ est la probabilité que le processus transite vers l'état s' quand l'action globale $a = \langle a_1, \cdots, a_n \rangle$ est effectuée dans l'état s ;
- $r() \ : \mathcal{S} \times \mathcal{A} \longrightarrow \mathbb{R}$ est une fonction de récompense. Effectuer l'action globale a dans l'état s donne une récompense de $r(s, a)$;
- b_0 est la distribution de probabilité initiale sur les états.

L'objectif de la thèse était de comprendre les propriétés et les contraintes de ce modèle dont la complexité de résolution est faramineuse puisque NEXP-difficile, comme l'a montré Bernstein et al. (2000). Nous nous sommes particulièrement intéressés aux liens entre les politiques locales des agents et la politique globale du système ce qui nous a aussi permis de proposer des algorithmes de résolution exacte des Dec-POMDP originaux. En général, ces algorithmes sont plus rapides que les algorithmes classiques de la programmation dynamique mais, sauf dans des cas bien particulier, moins rapides qu'un algorithme de recherche avant comme GMAA* (Oliehoek et al., 2008).

Comme pour les POMDP, nous avons travaillé pour des problèmes à horizon fini pour lesquels une solution optimale existe. Classiquement, les politiques des agents sont représentées par des arbres dont la profondeur est égale à l'horizon du problème. Un nœud de l'arbre représente l'action à exécuter et les arcs sont étiquetés par les observations possibles. Ainsi, suivant les observations

reçues après chaque action, l'agent parcourt l'arbre d'action en action. Dans l'exemple de la figure 4.2, deux agents d'ensembles d'actions respectifs $\{a_1, a_2\}$ et $\{b_1, b_2\}$, d'ensembles d'observations respectifs $\{u_1, u_2\}$ et $\{v_1, v_2\}$ contrôlent un Dec-POMDP d'horizon 2. Une trajectoire possible du processus global est ($\langle a_1, b_2\rangle$,$\langle u_1, v_1\rangle$, $\langle a_2, b_2\rangle$, $\langle u_1, v_2\rangle$, $\langle a_2, b_1\rangle$).

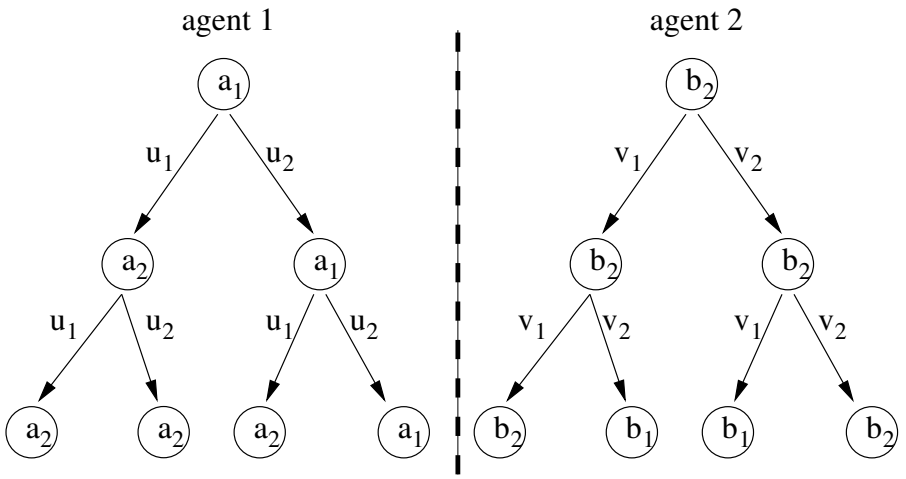


FIGURE 4.2 – **Arbres de politique pour deux agents**. Deux agents d'ensembles d'actions respectifs $\{a_1, a_2\}$ et $\{b_1, b_2\}$, et d'ensembles d'observations respectifs $\{u_1, u_2\}$ et $\{v_1, v_2\}$, contrôlent un Dec-POMDP d'horizon 2.

Le nombre de politiques possibles est beaucoup plus grand que le nombre de trajectoires possibles, car une même trajectoire peut être générée par plusieurs politiques. Pour un agent i, le nombre possible de politiques déterministes d'horizon T est :

$$|A_i|^{\frac{|O_i|^T-1}{|O_i|-1}} \tag{4.1}$$

alors que le nombre de trajectoires pour le même processus vaut :

$$\sum_{t=0}^{T-1} |O_i|^t |A_i|^{t+1} \quad = \quad \frac{|A_i|}{1-|O_i||A_i|}\{(1-(|A_i||O_i|)^T\}. \tag{4.2}$$

Une politique peut être représentée par l'ensemble des trajectoires qu'elle génère. Pour l'agent 1 de l'exemple précédent, sa politique génère les trajectoires suivantes : (a_1), (a_1, u_1, a_2), (a_1, u_2, a_1), $(a_1, u_1, a_2, u_1, a_2)$, $(a_1, u_1, a_2, u_2, a_2)$, $(a_1, u_2, a_1, u_1, a_2)$ et $(a_1, u_2, a_1, u_2, a_1)$. Mais les 4 dernières trajectoires sont suffisantes pour décrire entièrement la politique. Par contre, un ensemble de trajectoires donné ne définit pas forcément une politique, il doit satisfaire certaines contraintes. Par exemple, les deux trajectoires $(a_1, u_1, a_2, u_1, a_2)$, $(a_1, u_1, a_1, u_2, a_2)$ seraient incompatibles pour une même politique déterministe, car l'action choisie après la séquence (a_1, u_1) doit être soit toujours a_1, soit toujours a_2.

Pour résoudre un Dec-POMDP, nous avons proposé de nous appuyer sur une recherche des trajectoires individuelles qui doivent faire partie de la solution globale optimale plutôt que sur la recherche des politiques individuelles. La recherche s'effectue dans un espace beaucoup plus

réduit. L'ensemble des trajectoires individuelles doit néanmoins satisfaire de nombreuses contraintes car il faut s'assurer :

- que les trajectoires définissent bien des politiques individuelles ;
- que les trajectoires font bien partie de la solution optimale, ce que nous avons traduit par des contraintes en utilisant le fait que la solution optimale était forcément un équilibre de Nash du problème. Cette notion d'équilibre sera détaillée dans la section 4.3.

Ainsi, sans entrer dans des détails techniques et difficiles à exposer rapidement, nous avons montré qu'il était possible de résoudre un Dec-POMDP en résolvant un programme linéaire mixte entier, c'est-à-dire en cherchant les valeurs d'un ensemble de variables vérifiant des contraintes linéaires, certaines n'admettant que des valeurs entières, dans le but de maximiser une certaine fonction de ces variables. Un tel programme linéaire est donné dans la table 4.1. Ces variables sont des poids associés à chaque trajectoire, une politique individuelle étant définie par un ensemble de trajectoires de poids strictement positifs. Les détails de cette démarche, avec des programmes linéaires plus adaptés à certains cas, notamment au cas où seulement deux agents sont considérés, se trouvent dans nos publication (Aras et al., 2007a,b) et dans la thèse de Raghav Aras soutenue en 2008 (Aras, 2008).

Notre approche permet de résoudre des Dec-POMDP plus rapidement que les autres méthodes de la littérature qui s'appuient sur une adaptation au cas multi-agent de l'opérateur de la programmation dynamique (Hansen et al., 2004), mais l'algorithme de recherche avant heuristique GMAA* reste de loin le plus rapide (voir (Oliehoek et al., 2008; Aras and Dutech, 2009)). Une petite partie de ce gain en temps est due à l'utilisation d'un logiciel particulièrement optimisé pour résoudre les programmes linéaires, la représentation des politiques sous forme d'un ensemble de trajectoires étant à l'origine du reste du gain en temps. Le gain en temps de résolution, bien que significatif, ne rend pas les Dec-POMDP plus abordables pour autant. Du fait de l'utilisation de variables entières dans le programme linéaire, la solution est recherchée dans un espace dont la taille croit exponentiellement en fonction du nombre de trajectoires et, de ce fait, l'horizon des problèmes que l'on peut résoudre est le même que celui des algorithmes classiques, à savoir de l'ordre de 4 pour les problèmes qui ont servi à tester expérimentalement notre algorithme.

L'intérêt principal de ces travaux est une meilleure compréhension et représentation des liens entre les politiques individuelles des agents devant se coordonner, ainsi que des liens entre les politiques globales et individuelles. Une perspective alléchante concerne l'utilisation de notre représentation pour chercher des solutions approchées, par exemple en s'inspirant de techniques qui permettent déjà de s'attaquer à des problèmes d'horizon beaucoup plus élevé en utilisant l'opérateur de la programmation dynamique (Seuken and Zilberstein, 2007).

Un enseignement important de ces travaux concerne l'intérêt qui peut être donné aux trajectoires par rapport aux politiques. Une perspective serait de pousser cette démarche avec une connaissance imparfaite du modèle. L'idée serait de trouver quelques trajectoires permettant d'atteindre un état but ou de maximiser un critère de récompense, d'utiliser les outils que nous avons mis au point pour construire des politiques individuelles intégrant au mieux ces trajectoires et à raffiner ces politiques au fur et à mesure que l'expérimentation apporte de nouvelles connaissances sur le problème.

Le cadre de cette étude peut sembler assez éloigné de mes préoccupations à long terme car il n'y a pas ici d'apprentissage et une connaissance du modèle est nécessaire pour mettre en œuvre

Variables :

$x_i(h), \forall i \in I, \forall h \in \mathcal{H}_i$

$z(j), \forall j \in \mathcal{E}$

$$\text{Maximize} \quad \sum_{j \in \mathcal{E}} \mathcal{R}(\alpha, j) z(j) \tag{4.3}$$

subject to :

$$\sum_{a \in A_i} x_i(a) = 1, \quad \forall i \in I \tag{4.4}$$

$$-x_i(h) + \sum_{a \in A_i} x_i(h.o.a) = 0, \quad \forall i \in I, \forall h \in \mathcal{N}_i, \forall o \in O_i \tag{4.5}$$

$$\sum_{j' \in H^T_{-i}} z(\langle h, j' \rangle) = x_i(h) \prod_{k \in I \setminus \{i\}} |O_k|^{T-1}, \quad \forall i \in I, \forall h \in \mathcal{E}_i \tag{4.6}$$

$$\sum_{j \in \mathcal{E}} z(j) = \prod_{i \in I} |O_i|^{T-1} \tag{4.7}$$

$$x_i(h) \geq 0, \quad \forall i \in I, \forall h \in \mathcal{N}_i \tag{4.8}$$

$$x_i(h) \in \{0, 1\}, \quad \forall i \in I, \forall h \in \mathcal{E}_i \tag{4.9}$$

$$z(j) \in [0, 1], \quad \forall j \in \mathcal{E} \tag{4.10}$$

TABLE 4.1 – **Programme linéaire mixte pour résoudre un POMDP**. Voici un exemple de programme linéaire mixte entier permettant de trouver une solution à un POMDP. Pour chaque agent i, à chaque trajectoire d'observation-action h on associe une variable $x_i(h)$ qui vaudra 0 ou 1. Si elle vaut 1, cela veut dire que cette trajectoire doit pouvoir être générée par la politique optimale de cet agent. Les variables $z(j)$ sont de même associées aux trajectoires *jointes terminales* du processus. Les contraintes (4.4-4.5) assurent qu'il existe bien des politiques individuelles qui permettent de générer les trajectoires sélectionnées, ces politiques individuelles permettent de construire une *politique globale optimale* grâce aux contraintes (4.6-4.7). (Tiré de (Aras and Dutech, 2009))

nos divers algorithmes. Néanmoins, ces travaux effectués avec Raghav Aras représentent une étape importante dans notre compréhension des mécanismes présents au sein des systèmes multi-agents. De plus, ils capitalisent des connaissances acquises en se plongeant dans la littérature sur la théorie des jeux, capitalisation qui se traduit aussi dans les travaux que nous allons aborder maintenant dans un cadre beaucoup plus axé sur l'apprentissage.

4.3 Apprendre en s'appuyant sur la théorie des jeux

La théorie des jeux s'intéresse à des problèmes où plusieurs joueurs reçoivent chacun une récompense – on parle alors de *gain* – qui dépend du choix collectif. Chaque joueur est supposé rationnel et essaye donc de maximiser ses gains, avec une connaissance plus ou moins complète des données du problème. Chaque joueur a le choix entre plusieurs *stratégies* et la question posée est de savoir s'il existe un choix rationnel pour les joueurs. Le cadre le plus courant, car de nombreux jeux peuvent s'y ramener, est celui d'un jeu simple (nous dirions "composé d'un seul état") où les joueurs ne joueront qu'une fois. En considérant que chaque stratégie peut représenter une succession d'actions, voire une trajectoire d'actions et d'observations, on voit que de nombreux problèmes multi-agents peuvent se ramener à ce cadre faussement simple. Pour exposer les concepts principaux de la théorie des jeux, plaçons-nous dans le cadre de jeux à deux joueurs, que nous appellerons `agent 1` et `agent 2`.

La notion centrale de la théorie des jeux est celle de stratégies en équilibre qui correspond à un choix rationnel de la part des joueurs. Parmi ces différents types d'équilibres, l'équilibre de Nash est caractérisé par le fait que, individuellement, aucun joueur n'a intérêt à modifier sa stratégie, voir table 4.3. C'est un concept important car il a été montré que si les joueurs ont le droit d'utiliser des *stratégies mixtes*, c'est-à-dire de choisir leur stratégie en la tirant au hasard selon une distribution de probabilité sur l'ensemble de leur stratégies déterministes, alors tous les jeux admettent au moins un équilibre de Nash. Cet équilibre peut être formalisé de plusieurs manières :

- par le biais de la notion de *meilleure réponse*. Une stratégie d'un joueur `agent 1` est la meilleure réponse à une stratégie du joueur `agent 2` si elle assure, en moyenne, un gain maximal au joueur `agent 1`. Un équilibre de Nash est alors composé de stratégies qui sont toutes des meilleures réponses aux autres stratégies de l'équilibre ;
- par le biais de la notion de *regret*. Pour une stratégie du joueur `agent 2`, le regret associé à une stratégie du joueur `agent 1` est la plus grande différence entre le gain de cette stratégie et le gain d'une autre stratégie possible pour le joueur `agent 1`. Un équilibre de Nash est alors composé de stratégies dont le regret est nul.

En partant de l'algorithme classique de l'apprentissage par renforcement qu'est le *Q-Learning*, de nombreuses versions multi-agents utilisant les concepts de la théorie des jeux ont été proposés. Le *minimax-Q* de Littman (1994a) et *Friend-or-Foe Q-Learning*, sa version améliorée dans (Littman, 2001), permettent de résoudre des jeux à deux joueurs où ce que gagne l'un est perdu par l'autre, ce qui ne permet pas de faire collaborer des agents. L'algorithme *Nash-Q* de Hu and Wellman (1998a) est utilisable pour toute sorte de jeux mais sa convergence n'est pas garantie dans le cas général et ne cherche que des équilibres de Nash, qui peuvent parfois être sous-optimaux. Il faut de plus que chaque agent puisse connaître les actions et les récompenses de l'autre agent pour trouver les équilibres de Nash. C'est pour cela que l'algorithme *Correlated*

		agent 2	
		x_2	y_2
agent 1	x_1	-1 ; -1	-10 ; 0
	y_1	0 ; -10	-5 ; -5

TABLE 4.2 – **Jeu sous forme bimatricielle - Dilemme du prisonnier**. Dans ce jeu, x_1 et y_1 (resp. x_2 et y_2) sont les actions possibles pour `agent 1` (resp. `agent 2`). Dans chaque cellule de la bimatrice, le nombre de gauche représente le gain de l'`agent 1` et celui de droite le gain de `agent 2`. Le couple $[y_1; y_2]$ est un équilibre de Nash, alors que le couple $[x_1; x_2]$, qui rapporte plus aux deux agents, n'en est pas un. Ce deuxième "équilibre" pourrait être atteint en utilisant un *contrat* du type : "*Je promets de jouer x_1 (ou x_2) mais, si je suis le seul joueur à signer ce contrat, je jouerai y_1 (ou y_2)*".

Q-Learning de Greenwald and Hall (2003) fait apprendre aux agents des équilibres corrélés qui peuvent potentiellement être meilleurs que les équilibres de Nash, mais avec les mêmes restrictions concernant les connaissances nécessaires aux agents.

La famille d'algorithmes la plus aboutie est la famille des algorithmes du type *Win or Learn Fast* (WoLF) que l'on retrouve dans (Bowling and Veloso, 2002; Bowling, 2004) et qui s'inspirent de l'algorithme *policy iteration*. Les agents sont indépendants et n'ont pas besoin de connaissance sur les autres agents et ne perçoivent que le résultat de leur action. A chaque itération, la politique stochastique de chaque joueur est modifiée pour favoriser l'action qui paraît optimale mais le coefficient gouvernant l'amplitude de ce changement dépend du comportement actuel de l'agent. Si l'agent est en train de gagner, le coefficient est plus petit que si l'agent est en train de perdre. La convergence de cet algorithme vers un équilibre de Nash a été prouvée pour deux agents ayant chacun deux actions. L'expérience montre que ce résultat ne se généralise pas pour plus de deux agents ou plus de deux actions.

Pour aller plus loin, et conscient de la limite inhérente aux approche découplées démontrée par les travaux de Hart and Mas-Colell (2003), Raghav Aras et moi-même avons exploré différents types de couplage entre des agent permettant à des agents pourtant indépendants d'apprendre à jouer l'équilibre de Nash offrant le *meilleur* gain *global*. Une perspective de ces travaux était de déterminer la "quantité" de couplage, par exemple en terme d'information échangée entre les agents, en fonction du type de jeu. Nous avons plus spécialement étudié deux types de couplage dans un cadre d'apprentissage, c'est-à-dire dans des problèmes où les agents n'ont aucune connaissance du jeu. A chaque instant de décision, chacun des agents choisi une action parmi son ensemble d'actions et il reçoit une récompense qui dépend de l'action jointe. C'est une sorte de Dec-POMDP avec un seul état.

Dans un premier temps, nous avons considéré des agents couplés par l'échange de simples signaux binaires. Les observations des agents sont alors composées des signaux reçus par les autres agents et de leur perception de l'environnement. Les actions des agents sont composées d'une action sur le monde "physique" et de la décision d'envoyer un signal aux autres agents, en mode "broadcast". Aucune sémantique décidée *a priori* n'est associée à l'envoi d'un signal. Le but pour les agents est d'arriver à un état but commun en évitant certains états. Un agent

donné ne connaît qu'un sous-ensemble des états à éviter. Dans (Aras et al., 2004), nous avons évalué et analysé cette approche sur un problème académique où chaque agent pouvait modifier *un* chiffre d'un nombre à n chiffres, n étant le nombre total d'agents. Ce problème est illustré sur la figure 4.3.

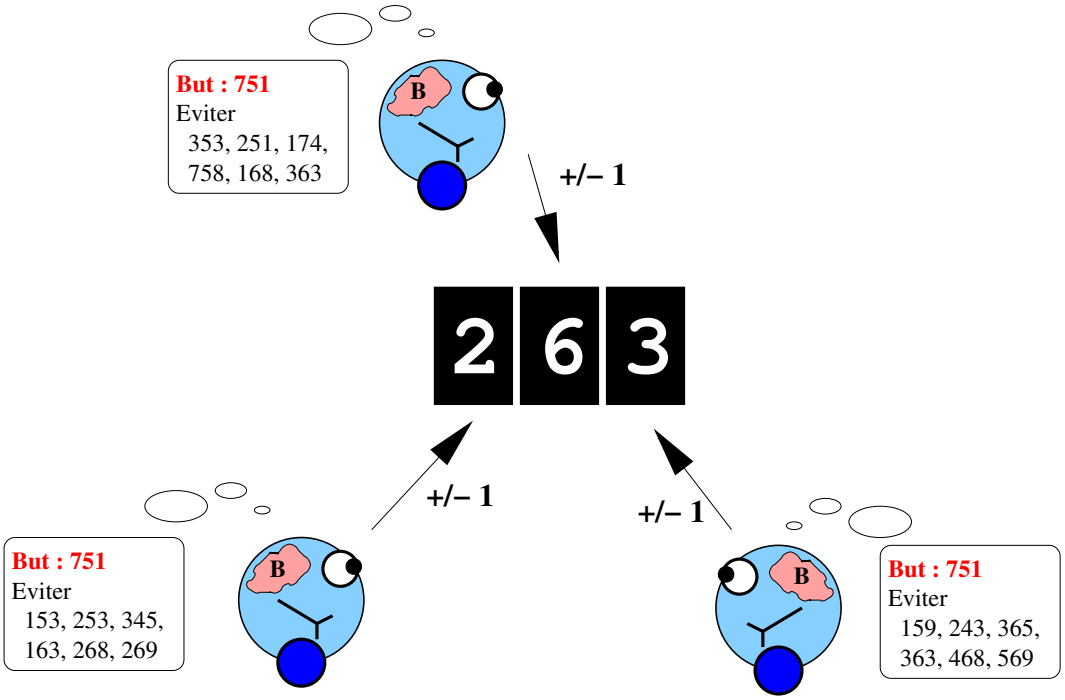


FIGURE 4.3 – **Le compte est bon**. Chaque agent est responsable d'un des chiffres et peut, à chaque action ajouter ou retrancher 1 à ce chiffre. La question posée est de savoir quelle est l'information *minimale* que les agents doivent échanger pour qu'ils parviennent ensemble à atteindre leur but (ici 751) sachant qu'ils ont chacun une liste de nombre à éviter.

Dans un deuxième temps, nous nous sommes inspiré de la notion de "dessous de table"[13] utilisée en théorie des jeux. L'idée est de permettre aux agents de jouer des équilibres corrélés qui, bien que parfois instables, peuvent apporter de plus grands gains globaux aux agents. Le gain global est une notion à préciser, dans notre cas nous avons choisi la somme des gains individuels des agents. Pour créer le couplage nécessaire permettant aux agents de se stabiliser sur ces équilibres, certains agents "promettent" de redistribuer une partie de leur gain aux autres agents. En fonction des jeux, certaines redistributions permettent quand même à tous les agents d'avoir un gain plus grand que le "meilleur" des équilibres de Nash et créant une sorte d'équilibre virtuel avec des gains globalement plus importants.

Puisque nous voulions un couplage le plus simple possible, nous avons proposé un mécanisme de *dessous de table virtuel* qui ne nécessite qu'un échange d'information minimal entre les agents. Le principe en est assez simple puisque chaque agent simule le fait qu'il reçoit, pour certaines de ses actions, une récompense virtuelle supplémentaire. Cette récompense est un moyen pour l'agent de s'inciter à jouer des actions qui lui paraissent localement sous-optimales mais qui

13. en anglais on parle de "side-payment"

peuvent s'avérer globalement optimales. Une fois les récompenses virtuelles fixées par les agents, ces derniers utilisent l'algorithme GIGA-WoLF (voir (Bowling, 2004)) pour trouver un équilibre de Nash virtuel. Les récompenses virtuelles sont maintenues pendant un certain temps avant d'être modifiées pour explorer de nouvelles configurations. Plus de détails sont disponibles dans (Aras et al., 2006).

Ces deux approches ont été testées sur des problèmes de coordination assez simples, allant de problèmes ne possédant qu'un seul équilibre de Nash à des problèmes où tous les équilibres de Nash étaient sous-optimaux. Ces problèmes étaient tirés aléatoirement parmi des familles de jeu en utilisant la bibliothèque GAMUT (Nudelman et al., 2004). Les performances de ces deux approches n'ont pas été très convaincantes ce qui n'est finalement pas très surprenant. Dans le premier cas, le signal ne véhicule que peu d'information et le couplage résultant est minimal. Les algorithmes d'apprentissage avec ce faible couplage ne se comportent guère différemment des algorithmes classiques, sauf dans quelques cas très spécifiques. C'est un premier pas qui nécessiterait d'y consacrer beaucoup plus de temps et d'énergie pour avoir une classification des différents types de jeux plus complète. Dans le deuxième cas, le problème d'apprentissage a finalement été déplacé : plutôt que d'apprendre une stratégie optimale, les agents essayent d'apprendre une configuration de récompenses virtuelles qui mène à un apprentissage d'une stratégie optimale. La dimension de l'espace d'états et d'actions des agents a été altérée, mais les problématiques fondamentales que doivent gérer les agents sont toujours les mêmes. L'algorithme, qui permet de résoudre de rares cas assez spécifiques, semble difficilement être utilisable dans un contexte plus général.

4.4 Une approche incrémentale

C'est par le biais d'une discussion autour de travaux pourtant antérieurs, puisqu'ils furent réalisé en 2000 pendant le DEA d'Olivier Buffet que j'ai encadré, que je voudrais clore ce chapitre sur l'apprentissage dans les systèmes multi-agents. Au cours de ces travaux, nous avons développé une démarche d'apprentissage incrémental en nous inspirant largement de travaux en psychologie sur l'apprentissage humain et notamment du concept de *shaping* (Skinner, 1953; Staddon, 1983). Ces travaux, bien que faisant appel à des outils parfois simplistes, me semblent dotés d'un fort potentiel et s'insérer parfaitement dans mes problématiques de recherche futures.

Nous considérons toujours un système composé d'agents simples, voire réactifs, qui doivent apprendre à résoudre une tâche globale en ne disposant que d'un point de vue local et limité sur leur monde. Pour pallier les difficultés de ce genre d'apprentissage, une méthode largement utilisée, sous bien des déclinaisons, est d'accompagner et de faciliter la tâche d'apprentissage. C'est une manière d'utiliser une expertise, une connaissance de la tâche pour faciliter l'apprentissage de l'agent. Transposée dans le cadre des MDP, cette démarche peut se traduire de différentes façons.

Il est d'abord possible d'adapter et de redéfinir la **fonction de récompense** pour guider l'agent qui apprend. Intuitivement, si l'on veut apprendre à un agent à trouver le chemin vers la sortie d'un labyrinthe, on peut soit le récompenser seulement quand il sort du labyrinthe (le problème d'apprentissage est ainsi facilement spécifié et défini) ou ajouter des récompenses à

certains points clefs du labyrinthe pour aider et "attirer" l'agent. Ces récompenses doivent être choisies avec soin pour que la solution du problème avec une fonction de récompense modifiée soit aussi une solution du problème original mais cette démarche est possible, voire fructueuse, comme le montrent des travaux sur des problèmes mono-agents ((Ng et al., 1999) et (Randlov and Alstrom, 1998)).

Une deuxième façon de procéder consiste à modifier les **probabilités de transitions** du problème. L'image classique est celle des roulettes que l'on rajoute sur un vélo pour apprendre aux enfants à faire du vélo. Les roulettes sont progressivement relevées au fur et à mesure que l'apprentissage progresse. Le résultat d'une action, et donc les probabilités de transition, diffèrent suivant que les roulettes sont présentent ou non. Cette idée a été appliquée sur des problèmes d'apprentissage mono-agent (Randlov, 2000).

Enfin, une troisième approche part du principe de présenter à l'agent des **tâches d'apprentissage** d'abord très simples et de les rendre de **plus en plus difficiles**, jusqu'à arriver à la tâche que l'on voulait initialement apprendre à l'agent. Il n'y a pas besoin de modifier la fonction de récompense (le but de l'agent) ou les probabilités de transition (la dynamique du problème). La principale difficulté est de quantifier les difficultés du problème présenté. L'idée est d'aider l'agent à estimer la valeur des couples *(état,action)* "proches" de ses buts et de faciliter la diffusion de ces estimations vers des états de plus en plus "éloignés". Les travaux de Asada et al. (1996) mettent en œuvre cette approche pour un agent où un seul état est récompensé et dans ce cas il est assez simple de trouver les états "proches" du but de l'agent.

Nous avons décidé d'utiliser cette troisième forme de *shaping* dans un contexte multi-agent car les deux premières sont moins compatibles avec la notion d'agent autonome. Modifier la récompense ou les fonctions de transitions peut impliquer d'avoir à modifier un agent "de l'intérieur". La troisième approche s'étend aussi plus naturellement aux systèmes multi-agents, ainsi que nous l'avons montré dans nos travaux (Buffet et al., 2001, 2007). L'approche que nous avons suivie augmente la difficulté de la tâche des agents selon deux axes, comme le montre la figure 4.4. D'une part, en travaillant avec un nombre minimal d'agents, nous avons défini une séquence de mini-problèmes de plus en plus complexes. D'autre part, le nombre des agents présents est progressivement augmenté, un nouvel agent étant en fait un clone d'un des agents qui a déjà appris, mais ce nouvel agent va ensuite apprendre de manière individuelle et indépendante. Notre approche a été validée expérimentalement sur un problème académique qui ne pouvait être résolu avec les algorithmes classiques d'apprentissage par renforcement.

Le fait de cloner les agents n'est pas compatible avec ma problématique générale qui considère des agents autonomes. Une solution, plus complexe à mettre en œuvre, consiste à entraîner les agents par petits groupes (2 à 2 dans le cas que nous avons étudié), avant de les mettre ensembles. C'est en tout cas un point qu'il faudrait examiner plus attentivement pour rendre compatible dans son intégralité cette approche du *shaping*.

Il est vrai que cette approche repose beaucoup sur l'intervention d'un expérimentateur "externe" qui est indispensable mais elle reste compatible avec mes buts à long terme. L'expérimentateur n'a pas besoin de toucher ou de manipuler les représentations ou les algorithmes internes de l'agent pour interagir avec lui, toutes les interactions sont en quelque sortes "externes" à l'agent. De fait, comme l'a noté Laud (2004), le *shaping* est une sorte de supervision légère de l'apprentissage par renforcement : on n'en change pas la nature mais on y apporte des connaissances et

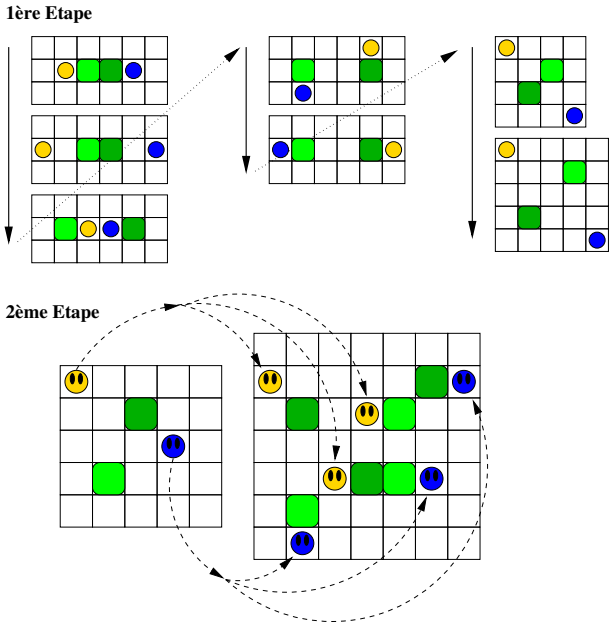


FIGURE 4.4 – **Shaping pour apprendre à fusionner des cubes**. Pour apprendre à des agents (ronds bleus ou jaunes) à pousser des cubes (carrés verts) l'un vers l'autre de manière à les "fusionner", nous avons fait travailler les agents sur des tâches de plus en plus complexes. Dans un premier temps, seuls deux agents sont utilisés. Ils sont placés régulièrement dans des positions de départ d'abord "proches" du but puis de plus en plus éloignées. Par exemple, dans la première situation (en haut à gauche de la 1ère Etape), les agents n'ont qu'à pousser les cubes. Mais dans la situation en haut au milieu, les agents doivent d'abord se positionner avant de pousser les cubes. Dans un deuxième temps, on augmente le nombre d'agents et la taille de l'environnement en partant de clones d'agents ayant déjà appris et en les laissant apprendre à gérer ces situations plus complexes. (Tiré de (Buffet et al., 2007)).

des compétences extérieures.

Une extension possible à ces travaux serait de les combiner avec nos méthodes de construction de nouveaux comportements par combinaisons de comportement plus simples déjà acquis par l'agent (voir section 3.5). Le *shaping* peut en effet guider l'agent et faciliter son acquisition de nouveaux comportements. Si l'on reprend la métaphore des roulettes, l'agent peut d'abord apprendre à pédaler et à se diriger avant d'apprendre à combiner ces comportements avec la contrainte de ne pas tomber. Quant à savoir si le *shaping* peut bénéficier de méthodes de construction de comportement, c'est une question plus ouverte et qui rejoint la problématique de l'automatisation du *shaping*.

Par automatisation du *shaping*, j'entends l'automatisation du processus de définition et l'élaboration du procédé de *shaping*, c'est-à-dire le choix des tâches qui seront présentées à l'agent ou auxquelles l'agent s'intéressera. Cette automatisation a pour but, soit de rendre le travail de l'expérimentateur plus simple, soit de permettre à l'agent d'être plus autonome et, en quelque sorte, de lui permettre de faire du méta-apprentissage. C'est en fait une question difficile dans

le cadre des MDP car il n'est pas facile de quantifier la difficulté d'une tâche ou de proposer une tâche "proche" d'une autre et plus simple à résoudre. Dans certains cas, la relation entre la "complexité" d'un modèle et la complexité de sa solution n'est pas très intuitive. Un exemple simple est celui de "la voiture sur la montagne" illustré figure 4.5. Il me semble ainsi que, du point de vue de l'agent, chercher directement quelles sont les séquences de tâches permettant d'apprendre la tâche globale revient finalement à apprendre le problème, voire même à résoudre une tâche plus dure et plus complexe que le problème lui-même. Ainsi, bien que la notion de "self-shaping" soit importante et aille dans la direction de mon but à long terme, je ne compte pas l'explorer plus avant pour l'instant.

Ces travaux posent aussi une autre question plus immédiate et non moins complexe qui concerne les motivations de l'agent. D'une manière ou d'une autre, un expérimentateur facilite l'apprentissage. Mais, finalement, qu'est-ce qui pousse l'agent à tirer parti de cette aide, à interagir avec l'expérimentateur ? Dans nos travaux, très limités, cette question ne se pose pas vraiment puisque l'agent a pour but de maximiser une fonction récompense que l'expérimentateur lui a choisie. Et l'expérimentateur tire parti de cette connaissance du fonctionnement intime de l'agent. Mais, dans le cadre plus vaste de l'agent se développant de façon vraiment autonome, il faudra résoudre plusieurs problèmes :

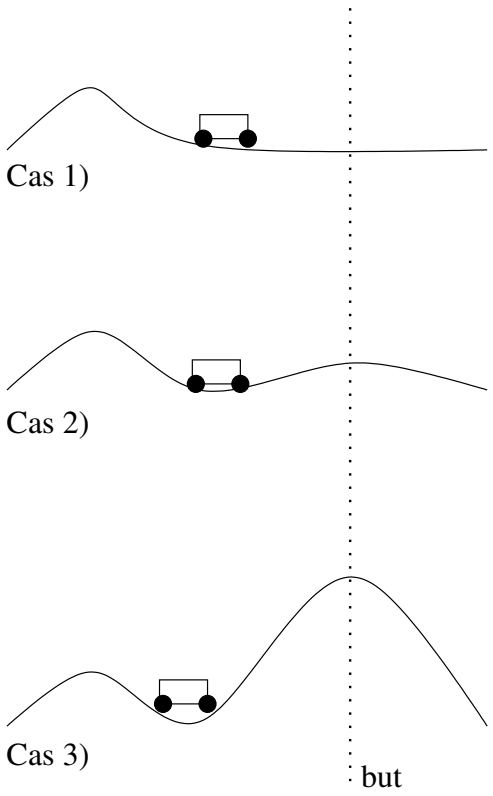
- l'agent ne sera pas intéressé par une seule et unique tâche ;
- son intérêt peut, et devra, changer avec le temps et son expérience ;
- bien que ne connaissant pas la motivation intime de l'agent, comment un expérimentateur "externe" (artificiel ou humain), pourra mettre en œuvre une sorte de *shaping* ?

Ces questions, et d'autres, ne pourront être occultées dans la suite de mes travaux. Nous y reviendrons donc dans la partie de ce manuscrit où je traiterai de mes perspectives de recherche.

4.5 Où en sommes nous maintenant ?

Essayons de tirer un bilan sur l'apprentissage par renforcement dans les systèmes multi-agents selon deux axes d'analyse. Intéressons-nous dans un premier temps à la problématique de l'apprentissage en tant que telle, c'est-à-dire que le problème fondamental à résoudre est celui de permettre à des agents indépendants d'adapter leurs comportements pour se coordonner afin de réaliser une tâche spécifiée globalement.

La formalisation et l'analyse de cette problématique, avec des outils comme les Dec-POMDP et la théorie des jeux, nous indique que c'est un problème extrêmement complexe. Il est illusoire de chercher un algorithme général et générique, surtout si l'on cherche à travailler avec des agents simples et réactifs, avec des capacités cognitives peu élaborées. Les approches efficaces s'appuient sur un couplage entre les agents. Une difficulté consiste à trouver un bon compromis entre la complexité du couplage entre les agents et la complexité de la tâche à apprendre. Plus il est évolué, plus un couplage va entraîner une certaine lourdeur, un gaspillage de ressources, aussi bien en termes de ressources de calcul, de mémoire, de bande passante, de temps de mise au point, *etc.* Nos travaux sur le sujet, mais aussi les nombreux travaux sur la communication dans les Dec-POMDP, sur les Dec-POMDP faiblement couplés, sur la co-évolution, sont des pas intéressants qui vont dans ce sens mais sont encore loin de nous donner une vision complète du problème. L'idéal serait de disposer d'une classification des problèmes d'apprentissage afin de



Le but est d'apprendre à la voiture d'aller s'arrêter à la position spécifiée par la ligne pointillée. La voiture a deux actions possible : accélérer vers la droite, accélérer vers la gauche. Dans le `cas 3)`, à cause de la pente, la voiture n'a pas assez de puissance pour aller directement vers son but, il lui faut d'abord accélérer vers la gauche, monter sur la colline pour y prendre de l'élan et ensuite seulement accélérer vers la droite pour atteindre le sommet de la montagne. Passer par les environnements plus simples (`cas 1)` et `cas 2)`) n'aide pas la voiture car ces environnements permettent directement d'aller vers le but sans s'aider de la colline. Quand on augmente progressivement la hauteur de la montagne but, la politique optimale change brusquement pour une certaine hauteur. Cette forme de *shaping* ne sert donc à rien.

FIGURE 4.5 – **La voiture sur la montagne**. Exemple d'une forme de *shaping* inutile.

pouvoir associer à chaque famille de problème le ou les types de couplage à mettre en place ou à apprendre pour que les agents puissent apprendre.

Comme d'autres, nous nous sommes intéressés aux apports possibles de la théorie des jeux pour cette problématique. Vu le nombre relativement restreint de travaux sur le sujet dans le domaine des MDP au sens large, il est évident que l'ensemble des résultats issus de la théorie des jeux n'a pas pu être exploité. Malgré tout, je suis moins enthousiaste qu'il y a quelques années sur le sujet. Cela vient en partie du fait que mon sujet de recherche principal n'est pas l'*apprentissage* dans les SMA, mais bien la compréhension des mécanismes sous-tendant la cognition. Cela vient aussi du fait que les sujets d'étude en théorie des jeux sont presques orthogonaux aux problématiques des chercheurs en intelligence artificielle, les uns s'intéressant principalement à l'existence de solution, les autres aux moyens de les trouver par apprentissage. C'est évidemment à nuancer : il y a de plus en plus de travaux qui font des ponts, mais la différence reste tangible.

D'une manière générale, je trouve que l'approche suivie jusqu'à présent et qui s'appuie sur des formalismes mathématiques comme les Dec-POMDP et la théorie des jeux, est peut-être trop "calculatoire", trop "mécaniste". J'entends par là que les agents sont dotés *a priori* de capacités de perception et d'action qui définissent et limitent les représentations qu'ils ont du monde. Dans cet espace abstrait, fini et limité, les algorithmes recherchent une solution en tirant parti de certaines régularités statistiques et stochastiques, ce qui est souvent une tâche ardue voire

trop complexe pour être réalisée en pratique. Des outils comme le *shaping*, les communications ou les "*side payments*" sont des façons d'aménager la recherche dans cet espace abstrait, mais ne changent pas la nature de la recherche qui reste *dématérialisée*. C'est en ce sens qu'il peut être utile de combiner *shaping* et "construction de comportement", mais bien d'autres idées sont à explorer. C'est dans cette direction que je compte continuer mes travaux sur l'apprentissage, en général et dans les systèmes multi-agents.

Le deuxième axe d'analyse de ces travaux dans les SMA concerne justement la problématique de l'agent autonome au sens de l'*embodiment*. Dans cette optique, le but premier des interactions est de faciliter l'apprentissage d'un ou plusieurs agents. On pense rapidement au rôle de professeur, de guide, de modèle, mais d'autres types de relations sont bien évidemment possibles. L'importance des interactions est confirmé par la nécessité de couplage entre les agents mais, comme je l'ai souligné en particulier lors de la discussion sur le *shaping* (section 4.4), de réelles difficultés sont à surmonter. Encore une fois, on peut poser cette question sous l'angle des capacités "innées" d'un agent artificiel à imiter ou tirer parti de ses interactions avec d'autres agents.

Enfin, j'ai déjà évoqué l'importance des motivations de l'agent, qui est un aspect que je compte développer plus avant. L'idée est de permettre des interactions pertinentes et utiles entre les agents. En ce qui concerne la motivation et la curiosité, des travaux ont déjà été effectués sur le sujet, notamment (Oudeyer et al., 2007). Il en va de même sur des aspects connexes comme l'élaboration de concepts commun à plusieurs agents ((Oudeyer and Kaplan, 2006)) ou l'apprentissage par imitation ((Price and Boutilier, 1999; Mataric, 1997)). Je compte pour ma part m'appuyer sur une meilleure compréhension des mécanismes biologiques de la motivation et du renforcement pour explorer des pistes différentes et, je pense, complémentaires. Comme je le détaille par la suite, les mécanismes biologiques ne sont pas sans liens avec les mécanismes multi-agents, notamment par leur aspect distribué et par l'importance de la notion d'émergence. Ces considérations m'amèneront à continuer à travailler aussi dans le domaine des systèmes multi-agents, comme je vais le préciser dans la partie finale de ce manuscrit.

5

Apprentissage par Renforcement et Robotique

Dans cette partie, je présente plusieurs travaux liés au domaine de la robotique. Par ces travaux, je montre que la robotique devient une préoccupation de plus en plus tangible et une composante de plus en plus présente dans mes recherches. Il me semble en effet utopique de prétendre travailler à comprendre les mécanismes de la cognition en se plaçant dans le courant de pensée de l'*embodiment* sans travailler avec des robots.

Les travaux présentés ici sont restés très modestes et ont plus servi à mettre en place un contexte que je compte exploiter par la suite. Je parlerai d'abord d'un travail sur une problématique intéressante mais uniquement réalisé sur un simulateur. Ensuite, j'aborderai deux expériences d'apprentissage par renforcement sur des véritables plateformes robotiques, l'une présente un apprentissage direct et l'autre un apprentissage indirect. Cette partie se termine sur les leçons que j'ai retirées de ces travaux.

5.1 Jeu de l'épervier

Le travail effectué lors du stage de Master de Louis Deflandre en 2006 peut être considéré comme préparatoire à l'activité robotique que j'ai mise en place par la suite (Deflandre, 2006). Bien que ces travaux aient uniquement été réalisés sur le simulateur robotique *Player/Stage*[14], ils me semblent importants à signaler ici car ils sont assez emblématiques d'un certain type de problématique robotique que je compte explorer dans le futur.

Ainsi qu'illustré par la figure 5.1, le problème, non-markovien, posé à l'agent robotique est le suivant. L'agent, appelé agent chasseur ou agent épervier, est placé avec d'autres robots dans un environnement sans obstacle. Les autres robots sont des "proies" mais, à un moment donné,

14. http ://playerstage.sourceforge.net

une et une seule proie peut bouger. Le but de l'agent est alors de rattraper et toucher la proie qui bouge. Les perceptions de l'agent chasseur sont :

- un retour très fruste (on/off) sur le fait que l'agent est lui-même en train de se déplacer ;
- une caméra linéaire à 360 degrés discrétisée permettant de distinguer les autres robots de l'environnement sous forme de "blobs" de couleur. Par le biais de cette caméra, l'agent connaît les positions angulaires approximatives des autres robots. En fonction de la "largeur" apparente des autres robots, l'agent chasseur peut aussi se faire une idée très approximative de la distance des autres robots. Cette perception reste très bruitée et imprécise.

A chaque instant de décision, l'agent chasseur a le choix entre 3 comportements très simples, à savoir :

- s'arrêter ;
- changer son point de focalisation sur un autre "blob" visible par l'agent. Le point de focalisation va essayer de suivre la cible choisie, même si elle bouge. Le processus de focalisation d'attention est réalisé en s'appuyant sur les travaux de l'équipe Cortex sur l'attention visuelle avec des cartes neuromimétiques mettant en œuvre la CNFT ("*Continuous Neural Field Theory*") (Rougier and Vitay, 2005) ;
- se mettre en chasse vers la cible sur laquelle il est focalisé.

La tâche est délicate à réaliser car, outre les problèmes inhérents à ses perceptions limitées et bruitées, l'agent est confronté à un problème non-markovien et non-stationnaire car il ne sait pas quand ou comment les autres agents vont bouger. De plus, quand le chasseur est immobile, il doit centrer son intérêt sur la cible qui bouge le plus mais quand il est en chasse, il doit au contraire s'intéresser à la cible qui, en général, bouge le moins de son point de vue.

Je trouve la problématique de l'agent épervier intéressante car :

- l'agent doit coupler des données proprioceptives – le fait qu'il soit en mouvement ou pas – avec des données extéroceptives – les positions relatives des autres agents – pour mener à bien sa tâche ;
- l'agent doit faire avec des perceptions limitées, brutes et bruitées ;
- l'agent est face à un problème non-markovien ;
- si l'agent apprend à poursuivre et toucher la cible mobile, alors il a aussi implicitement catégorisé les cibles potentielles en "cibles à poursuivre" et "cibles à ignorer".

Dans le cadre du stage de Louis Deflandre, l'agent épervier était doté d'une perception adaptée au problème puisque la cible vers laquelle l'agent était focalisé était catégorisée en terme de la donnée assez fruste de sa vitesse de déplacement relatif. Ces perceptions adaptées escamotent le point le plus dur du problème et le plus intéressant du point de vue de l'*embodiment* car on peut penser que l'agent n'a plus qu'à faire le lien entre le déplacement apparent de la cible et son propre déplacement pour détecter la cible à poursuivre. Mais, même ainsi, la tâche s'est révélée très ardue à apprendre malgré l'utilisation d'un algorithme de type SARSA(λ) pour essayer de pallier certaines difficultés pressenties. Les causes principales de cet échec relatif sont :

- rareté des récompenses non-nulles reçus. L'agent épervier ne réussit que très rarement à immobiliser une bonne cible, dès lors il est très rarement récompensé. L'emploi de traces d'éligibilité dans SARSA(λ) n'a pas réussi à compenser ce fait (cf. (Sutton, 1996), chapitre 7) ;
- problèmes d'occlusion et surtout de fusion dans la vision des cibles. Quand la cible suivie passe derrière ou devant une autre cible, il devient difficile pour le système d'attention visuelle dont est doté l'agent épervier de suivre la bonne cible à coup sûr ;
- perceptions trop frustes et trop bruitées. La combinaison de ces deux composantes fait qu'il est parfois difficile pour l'agent épervier de catégoriser une cible pourtant effective-

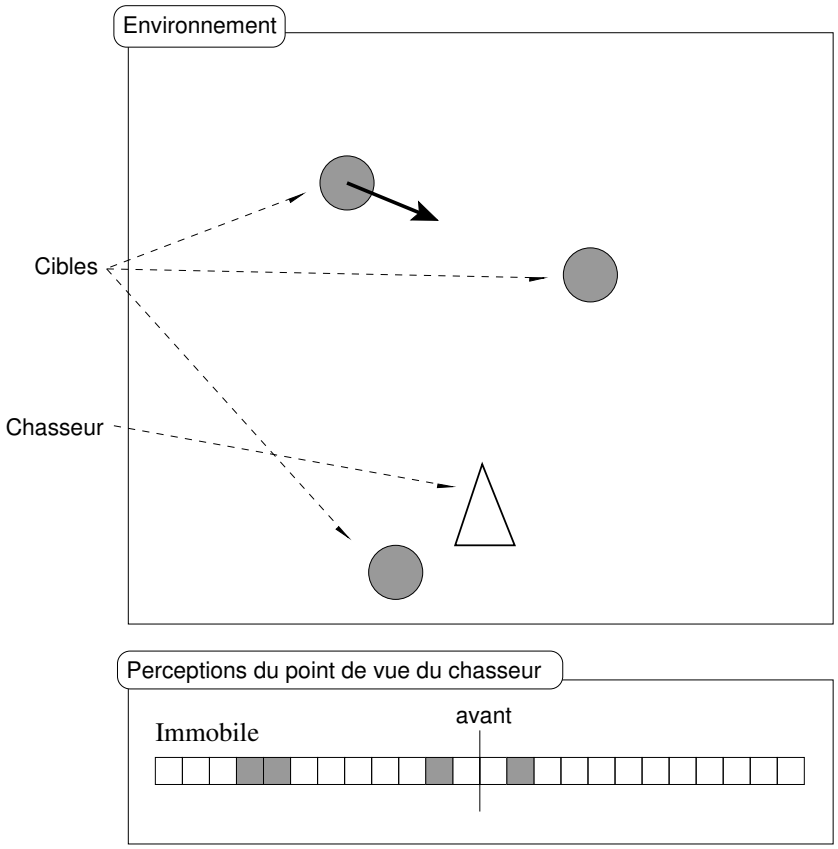


FIGURE 5.1 – **Problématique de l'agent épervier**. Un agent chasseur doit essayer de capturer *la* cible qui est en mouvement. Ses perceptions sont composées d'une caméra 360 degrés linéaire et d'une information binaire sur le fait qu'il est lui-même en mouvement ou non. Dans le cas présent, il détecte 3 "blobs" correspondant aux 3 cibles, dont l'un est plus gros car la cible est plus proche. De plus, le chasseur est `immobile`.

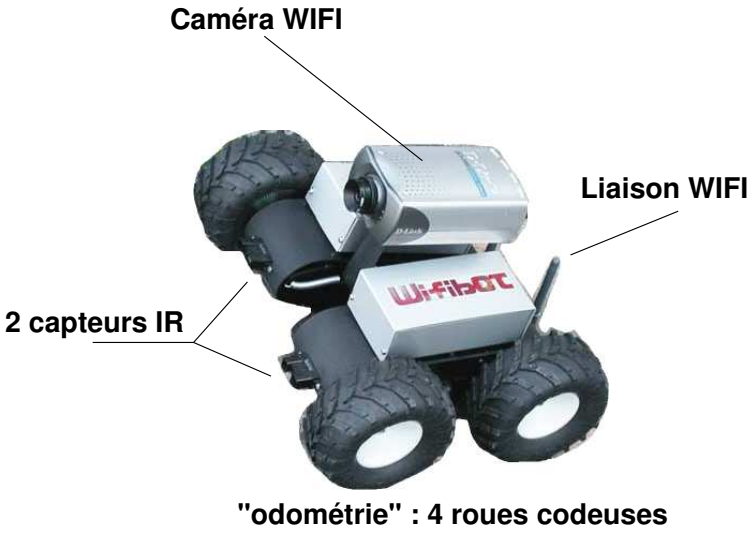


FIGURE 5.2 – **Robot WifiBot**. Plateforme expérimentale de robot mobile, le WifiBot est équipé de peu de capteurs : une caméra wifi et deux capteurs infrarouges placés à l'avant.

ment immobile comme étant immobile. De même, il est difficile de bien appréhender le déplacement apparent d'une cible.

Cette première approche de la robotique, aussi imparfaite et frustrante qu'elle ait pu être, s'est tout de même révélée intéressante en terme de réflexion sur le traitement des perceptions d'un robot, sur les perspectives que pourraient ouvrir une méthode plus incrémentale de l'apprentissage sur ce genre de problème et sur des problématiques robotiques de bas niveau.

5.2 Apprentissage par renforcement direct

C'est dans l'optique de vraiment traiter des aspects "bas-niveau" sur une véritable plateforme robotique que j'ai travaillé avec Nicolas Beaufort sur des WifiBots [15] (voir figure 5.2). Le but de ce travail était de mettre en œuvre un algorithme d'apprentissage par renforcement direct pour apprendre au robot à effectuer des tâches très simples.

Ce faisant, nous avons voulu explorer plusieurs problématiques liées à l'apprentissage par renforcement en environnement réel :

- **Comment gérer un espace d'état continu ?** Le robot évolue dans un monde intrinsèquement continu. Nous n'avons pas voulu artificiellement discrétiser les perceptions du robot mais, au contraire, chercher des moyens pour que l'apprentissage puisse se dérouler avec des perceptions, sinon "brutes", mais au moins continues. Comme décrit en section 2.4.2, une large littérature abonde sur le sujet dans le cadre de la "planification" mais nous avons ici la contrainte supplémentaire que le robot ne connaît pas la dynamique de son environnement ;

15. http ://www.wifibot.com

– **Comment apprendre efficacement ?** Dans le cadre d'un robot, le coût temporel d'une interaction avec l'environnement est un facteur important à considérer. Le robot ne peut se permettre d'effectuer des dizaines de milliers d'interactions pour apprendre si chaque interaction dure quelques secondes. Un de nos objectifs est donc de proposer des algorithmes d'apprentissage "efficaces", qui tirent le meilleur parti des interactions avec l'environnement pour apprendre avec le moins d'interactions possible. Cette problématique est présente depuis fort longtemps dans la communauté de l'apprentissage par renforcement (Thrun, 1992; Moore and Atkeson, 1993; Kearns and Koller, 1999; Li et al., 2008) et notre objectif était de l'explorer conjointement avec un environnement continu.

Nous nous sommes largement inspirés des travaux de (Smart and Kaelbling, 2002; Smart, 2002) mais sur une tâche différente et avec une fonction de récompense moins *ad hoc*. La tâche soumise au robot est d'apprendre à s'approcher d'une cible aisément identifiable et de s'arrêter devant elle. Les perceptions du robot sont constituées de la direction relative de la cible et de sa largeur apparente, quand la cible est visible. Ces informations sont extraites de l'image donnée par la caméra embarquée par le robot. A chaque instant de décision, le robot doit choisir entre 4 actions abstraites : avancer pendant 1 seconde, tourner sur place à gauche ou à droite, ne rien faire. Après chaque action, le robot reçoit un signal de récompense qui est nul sauf quand la cible est en face de lui et que sa taille est dans un certain intervalle, ce qui occasionne alors une récompense positive.

Pour gérer le fait que l'espace d'état du robot est continu – l'angle et la largeur perçus de la cible sont des valeurs continues –, le robot cherche à approcher la fonction de valeur en chaque couple (état, action) en s'appuyant sur une régression linéaire locale pondérée (Cohn et al., 1996; Atkeson et al., 1997). La valeur d'un état donné est le résultat d'une régression réalisée sur les valeurs des états proches que le robot a déjà explorés, comme l'illustre la figure 5.3. Quand la valeur d'un état est mise à jour, les valeurs des états proches sont aussi mises à jour. Pour accélérer l'apprentissage, le robot est manuellement guidé depuis quelques positions de départ prises au hasard vers son but, ce qui lui permet de mémoriser certains états et de calculer rapidement leur valeur.

Cette étude sur une plateforme robotique m'a permis d'avancer sur plusieurs points. J'ai pu mieux cerner les questions que je voudrais aborder dans un futur plus ou moins proche. Il me faut d'abord signaler qu'avec une dizaine de trajectoires exemples, le robot est capable de naviguer vers son but tant qu'il ne s'éloigne pas trop des exemples. En fait, la régression locale pondérée permet effectivement d'exploiter une approximation assez bonne de la fonction de valeur pour conduire le robot à son but quand il part d'une position proche d'une position déjà rencontrée. Par contre, le pouvoir de généralisation de l'algorithme n'est pas très bon et le robot se retrouve rapidement dans des situations où il est trop éloigné de situations déjà rencontrées pour savoir comment agir au mieux : la valeur approchée de chaque action en cette nouvelle situation est nulle. En outre, les perceptions du robot le placent de fait dans un cadre non-markovien, comme le montre la figure 5.4. Combiné avec le manque de généralisation de l'algorithme, le cadre non-markovien a rendu l'apprentissage assez difficile, surtout dans les situations où le robot perd la cible de vue, avec, finalement, des résultats plutôt moyens.

Ces travaux amènent des questions sur le système perceptif du robot. Nous n'avons pas voulu utiliser les images brutes données par la caméra comme entrée directe du système de décision du robot. Cette information nous paraissait à la fois trop riche et trop difficilement exploitable. L'être humain, à l'image de l'animal, possède un système de vision effectuant une sorte de

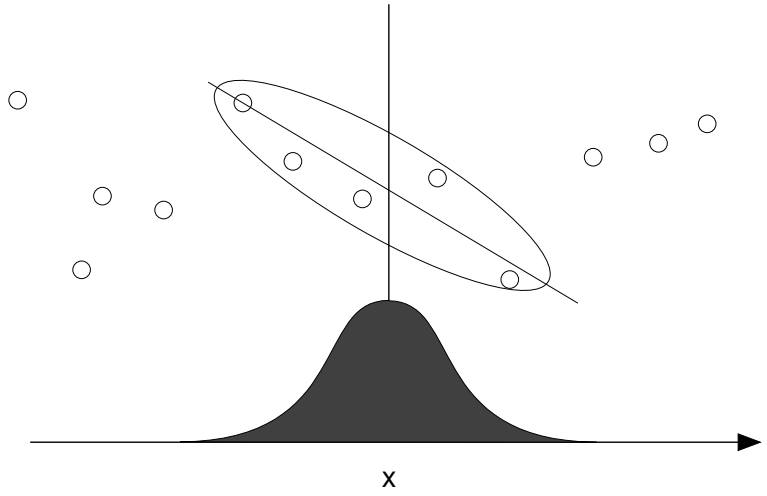


FIGURE 5.3 – **Régression locale pondérée**. Pour calculer la valeur du point x, les points connus proches de ce point sont pondérés par une fonction noyau (par exemple la gaussienne grisée). On calcule ensuite une régression sur ces points pondérés.

prétraitement des entrées visuelles (détection de vitesse, de couleur, de segments, *etc*). Sans entrer dans la question de savoir ce qui est acquis ou inné, sans même s'intéresser aux aspects beaucoup plus complexes de la vision, nous nous sommes demandés quelle part de prétraitement semblait adaptée à notre tâche pour qu'elle soit à la fois réalisable mais pas non plus trop élémentaire. Le choix que nous avons fait doit être repensé car le prétraitement dont nous avons doté me semble à la fois :

- **trop sophistiqué**. Outre le fait que les traitements informatiques nécessaire à produire l'information de `taille` et `angle relatif` de la cible sont assez complexes, la précision des perception est trop élevée par rapport à la tâche et à la précision des capteurs. En ce qui concerne l'`angle`, il est donné comme la position en pixels du centre mesuré de la cible sur l'image fournie par la caméra, soit un nombre entre 0 et 639. Il en va de même pour la `taille` de la cible. Compte-tenu de la précision avec laquelle la cible est repérée dans l'image et du fait qu'il faut ensuite faire un calcul approché de la fonction de valeur dans l'espace d'état donné par (`taille`, `angle relatif`), une telle précision et un tel intervalle de variation ne sont pas nécessaires et compliquent sans doute la tâche ;
- **trop réduit**. D'un autre côté, les perceptions dont est doté l'agent le placent devant une tâche non-markovienne que les algorithmes utilisés ne garantissent pas de résoudre, même dans le cas favorable où les espaces considérés sont discrets. Une perspective serait d'augmenter les modalités et les canaux de perception redondants avec, par exemple, des informations odométriques, des informations de distance issues de capteurs infrarouges, une donnée sur le déplacement ou la vitesse de déplacement de la cible.

Ces questions ne sont pas faciles à traiter et nous allons voir qu'elles réapparaissent sous une forme légèrement différente dans les travaux suivants sur l'apprentissage indirect.

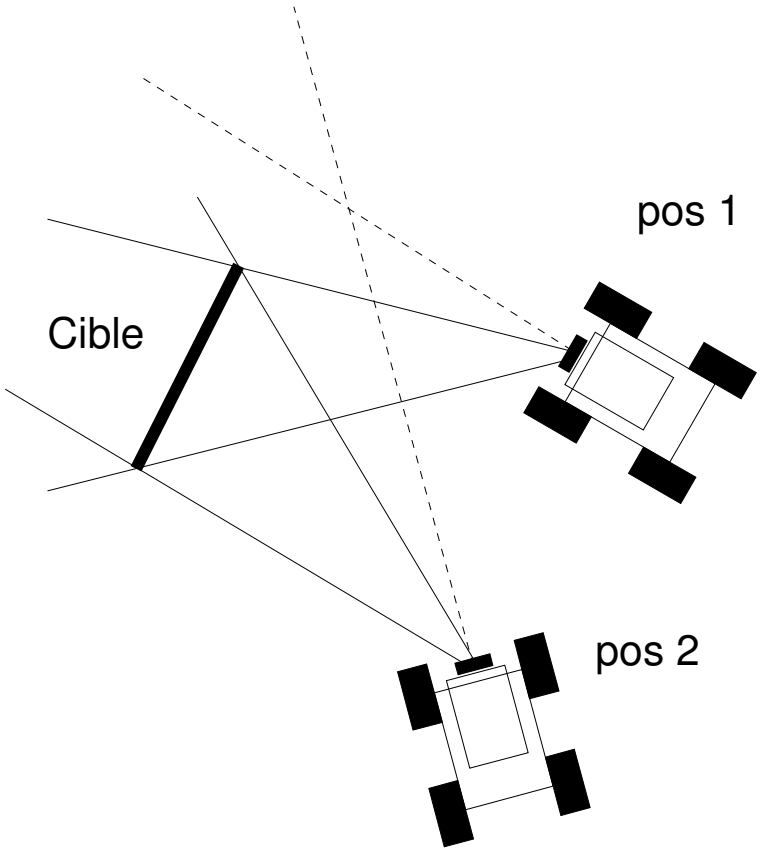


FIGURE 5.4 – **Une problématique non-markovienne**. Dans les deux positions indiquées sur la figure, la "perception" de la cible est la même : même largeur perçue et même angle relatif. Mais si le robot avance, dans un cas – `pos 1` –, la cible va "grossir" alors que dans l'autre – `pos 2` – elle va "diminuer" de taille. Avec ces perceptions et ces actions, la problématique que doit résoudre le robot n'est pas markovienne.

5.3 Apprentissage par renforcement indirect

Beaucoup des concepts du travail précédent ont été réutilisés sur une plateforme robotique différente avec, comme principal but, de mettre en œuvre une approche d'apprentissage par renforcement *indirecte* dans un cadre robotique. Comme je vais le préciser plus bas, il s'agit pour le robot, qui évolue toujours dans un monde continu, d'apprendre une fonction de transition lui permettant *ensuite* de planifier ses déplacements. Mais plusieurs objectifs secondaires sont attachés à ce travail réalisé lors du stage de Master "recherche" de Nicolas Beaufort (Beaufort, 2009). Nous voulons en effet :

- explorer une approche plus proche du paradigme de "l'*embodiment*" en laissant au robot la possibilité d'utiliser son environnement pour faciliter sa tâche de navigation ;
- améliorer et stabiliser les modules et bibliothèques robotiques "bas niveau" que nous pourrons réutiliser dans nos futurs travaux de recherche. C'est un objectif plus technologique mais essentiel pour la suite ;
- enfin, je considère ce travail et le travail précédent comme des étapes préliminaires pour mener une étude plus conséquente sur les possibilités, le potentiel, les avantages et les inconvénients liés à un apprentissage utilisant en parallèle une approche directe et indirecte. Comme postulé par Daw et al. (2005), ces deux voies d'apprentissage seraient à l'œuvre en parallèle chez les humains, ce qui suscite des questions sur le pourquoi et le comment d'un tel système. En collaboration avec l'équipe d'Alain Marchand du Centre de Neurosciences Intégratives et Cognitives (CNIC) de Bordeaux, nous voulons explorer ces questions sous trois angles différents (neurosciences, robotique, modélisation mathématique).

La tâche servant de support à ce travail sur l'apprentissage indirect est une tâche classique de navigation. Le robot doit apprendre à se déplacer pour rejoindre un but dans un environnement qui lui est *a priori* inconnu et parsemé d'obstacle. La photo de la figure 5.5 présente une vue aérienne de l'environnement dans lequel évolue le robot.

Les robots utilisés dans ce travail sont des Khepera III [16] qui sont décrits dans la figure 5.6. En s'appuyant sur ses capteurs infrarouges, le robot est doté d'un comportement d'évitement d'obstacle inspiré des véhicules de Braitenberg : un obstacle détecté à gauche (resp. droite) implique un virage sur la droite (resp. gauche) (Braitenberg, 1984). Ce comportement est toujours actif sur le robot. La portée des capteurs étant assez faible (entre 5cm et 20cm), ce comportement n'influence les déplacements du robots que lorsqu'il est très proche d'obstacles.

A ce comportement sous-jacent s'ajoute le comportement de navigation proprement dit. Comme précédemment, à chaque instant de décision, le robot doit choisir entre 4 actions : avancer pendant 1 seconde, tourner sur place à gauche ou à droite, ne rien faire. Pour prendre sa décision, le robot ne dispose que de ce qu'il a appris et de la perception de sa localisation dans le monde, localisation qui est donnée grâce à une caméra située au dessus de la zone d'évolution du robot. La localisation est constituée de la position (x, y) du robot et de son orientation θ relativement à une direction fixe et arbitraire. La localisation est imprécise, notamment quand le robot se trouve à la périphérie de sa zone d'évolution, et bruitée, surtout en ce qui concerne l'orientation.

16. De la société K-Team, http ://www.k-team.com

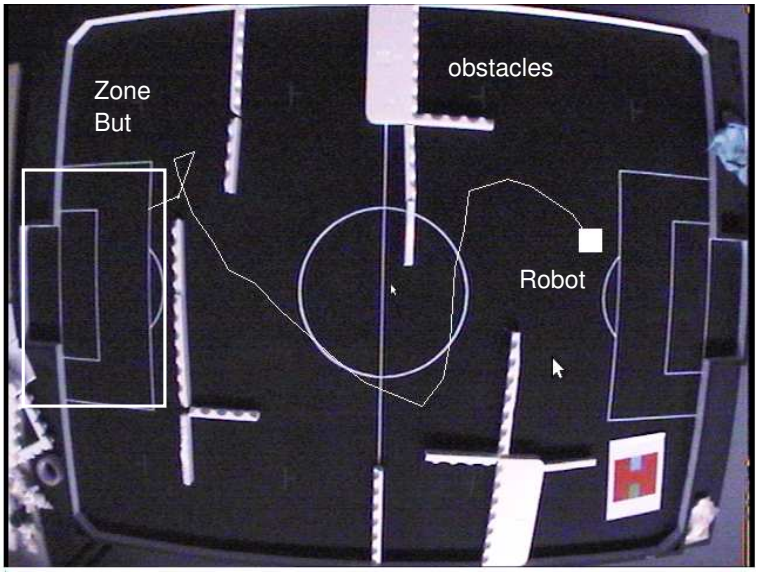


FIGURE 5.5 – **Une tâche de planification classique**. Cette vue aérienne montre l'environnement du robot qui doit se déplacer dans un environnement délimité par des "murs" extérieurs et parsemé d'obstacles (blocs de polystyrène). L'agent doit apprendre à gagner la zone but marquée par un rectangle blanc sur la photo (mais non apparent dans l'environnement). Sur la photo nous avons aussi fait apparaître la trajectoire planifiée par le robot symbolisé par le carré blanc, trajectoire qui est clairement non-optimale.

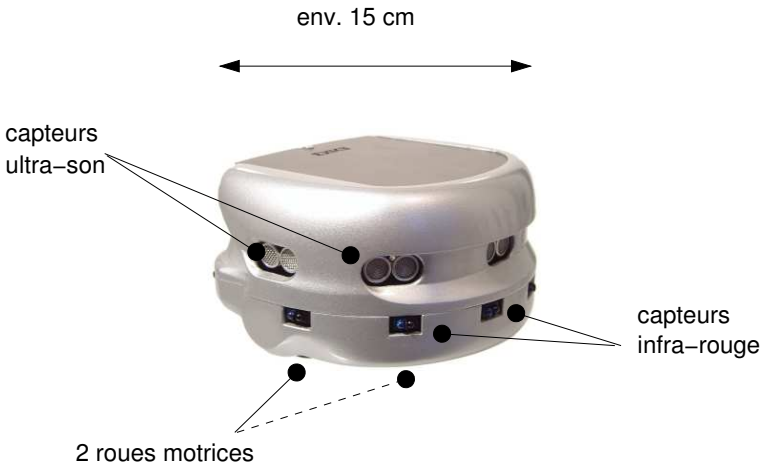


FIGURE 5.6 – **Robot Khepera III**. Dans ce travail, nous avons essentiellement utilisé les capteurs infrarouges situés sur le pourtour du robot.

L'approche que nous avons suivie emprunte beaucoup au travail précédent en robotique sur l'apprentissage direct en terme d'outils d'approximation. L'apprentissage est amorcé en montrant au robot quelques trajectoires le menant à son but. De ces trajectoires, le robot apprend une approximation de la fonction de transition du monde en s'appuyant sur des méthodes de plus proches voisins. Il lui est alors possible, quelle que soit sa localisation dans l'environnement, de planifier "en-ligne" une trajectoire le menant à son but en utilisant une version adaptée de l'algorithme RTDP (Barto et al., 1995).

Ce travail, encore une fois préliminaire et préparatoire à une exploration plus poussée de l'apprentissage par renforcement en robotique, est perfectible sur bien des points. A mon sens, et comme précédemment, c'est au niveau des perceptions qu'il y a le plus à redire. L'"état" perçu par le robot dépend de sa localisation qui est le produit d'une perception "externe" au robot. De plus cette localisation s'appuie sur des techniques de traitement d'image qui sont très *ad hoc* et qui ne donnent pas des informations *situées* mais belles et bien *globales*. Nous sommes arrivés à cette entorse indéniable au courant de pensée de l'*embodiment* car notre priorité était de tester l'apprentissage de l'approximation des transitions et, pour cela, le robot devait avoir une notion de son "état", de sa situation dans le monde. La caméra extérieure était le moyen le plus simple de disposer d'information non-ambigüe sur la situation du robot. D'autres solutions plus autonomes sont possibles, notamment en couplant des données odométriques avec des données issues des capteurs à ultra-son, et une question qui se pose est de savoir s'il est impératif de passer par des perceptions plus situées et internes au robot avant de continuer à étudier l'apprentissage d'une approximation des transitions ou si on peut d'abord approfondir cet apprentissage indépendamment de la "perception". N'est-on pas en train d'escamoter une partie d'un problème fondamental en terme d'*embodiment* ?

Car d'autres questions, parfois aussi fondamentales mais plus focalisées sur l'approximation, subsistent. Les trajectoires actuellement planifiées par le robot sont sous-optimales. Comment améliorer ces trajectoires ? Faut-il plus d'exemples, adapter certains paramètres en ligne, ajouter des comportements spécifiques ? L'optimalité est-elle vraiment l'objectif à atteindre ? Comment améliorer les performances du robot si on ne dispose pas d'un "critère" que l'on cherche à optimiser ? Nous voudrions aussi creuser l'influence de l'amorçage, tant en terme de quantité de trajectoires exemples fournies au robot qu'en terme de répartition de ces trajectoires, sur la qualité de l'apprentissage. Il nous semble aussi important de doter le robot d'une certaine "curiosité" le poussant à aller explorer des situations qu'il n'a pas encore rencontrées, ce qui est en fait une question à tiroir des plus difficiles : comment savoir qu'une situation est non-explorée, qu'elle peut être utile, que cela ne se fait pas trop au détriment de l'objectif principal du robot ?

Parmi les points positifs de ce travail, je voudrais en citer trois.

- **Socle technique pour de futurs travaux**. Parallèlement aux questions scientifiques, ce travail de recherche avait aussi pour but de préciser et de consolider notre plateforme robotique, tant du point de vue matériel que logiciel. Sans entrer dans le détail, à l'issue de ces travaux, nous disposons d'une plateforme où les travaux ultérieurs pourront être conduits en s'affranchissant largement des problématiques techniques "bas-niveaux" qui ne relèvent pas de notre centre d'intérêt. Ce socle technique s'appuie sur une bibliothèque écrite en C/C++ et la possibilité d'utiliser un langage de plus haut niveau pour programmer directement les robots au niveau comportemental (le langage URBI[17]) ;

17. De la société GOSTAI, http ://www.gostai.com

- **Perceptions bruitées et imprécises**. La localisation du robot est imprécise et bruitée. Même quand le robot est immobile, la position du robot est donnée avec un bruit de l'ordre de 1% mais par contre l'incertitude au niveau de l'orientation est de l'ordre de ±30 degrés, ce qui est assez considérable. De plus, plus le robot s'éloigne du centre de son aire de déplacements, plus sa localisation est imprécise car l'image de la caméra est déformée à la périphérie. Malgré tout, le robot est globalement capable de naviguer jusqu'à son but, avec quelques zones de son environnement d'où il ne peut pas "sortir". Ce résultat est néanmoins intéressant étant donné le côté "fruste" des perceptions utilisées ;
- **Bouclage fort avec l'environnement**. La superposition de deux niveaux de comportements (évitement d'obstacle bas niveau et décision haut niveau) dans une sorte d'architecture de subsomption (voir (Brooks, 1986)) mène parfois à un bouclage fort entre le robot et son environnement. Ainsi, dans des situations analogues à celle de la figure 5.7, l'architecture du robot lui permet parfois de vraiment tirer parti de l'environnement pour naviguer : c'est en "exploitant" son comportement d'évitement d'obstacles que le robot se dirige vers son but alors qu'il aurait aussi pu décider de tourner sur place puis d'avancer, ce qui peut sembler moins efficace en terme de nombre de déplacements pour rejoindre le but mais plus facilement reproductible. Mais, si on tient compte de l'incertitude potentielle sur l'orientation du robot, la technique consistant à laisser l'évitement d'obstacle orienter le robot vers sa cible est en fait beaucoup plus robuste.

5.4 Quelles leçons en retirer ?

Nos travaux, s'ils s'inscrivent dans un intérêt croissant pour les méthodes probabilistes et les méthodes d'apprentissage dans la communauté robotique, sont assez éloignées des recherches de pointe même s'ils en partagent les concepts. Comme le montrent en effet un numéro spécial très récent de la revue "*Automous Robot*" dédié à l'apprentissage ou les "*Workshops*" à venir sur l'apprentissage et la robotique, le cadre formel de l'apprentissage par renforcement permet de poser des problèmes de génération et de contrôle de mouvement comme des problèmes d'optimisation paramétrique ou d'approximation linéaire, non-linéaire, bayésienne, *etc.* Le principe est de disposer d'échantillons de trajectoires, générées par des exemples humains ou à partir d'un contrôleur à améliorer, d'en déduire une approximation de la fonction de valeur et d'en déduire un meilleur contrôleur (Riedmiller et al., 2009) ou de chercher directement à améliorer le contrôleur existant (Vlassis et al., 2009; Martinez-Cantin et al., 2009). Ces méthodes, qui s'appuient sur des techniques efficaces et rapide d'approximation, permettent d'apprendre rapidement à attraper un ballon, marcher, faire voler un hélicoptère sur le dos ou frapper un balle avec une batte de baseball.

Nos résultats sont loin de pouvoir rivaliser avec l'efficacité de ces méthodes et les résultats impressionnants qu'elles permettent d'obtenir. Il me faut néanmoins nuancer ce constat de deux façons. D'une part, nous avons travaillé avec des perceptions très frustes, bruitées et qui peuvent même conduire à des problématiques non-markoviennes alors que les travaux que je viens de citer disposent de perceptions plus élaborées, plus précises, plus adaptées et s'efforcent de placer le robot devant une problématique markovienne. D'autre part, les fonctions de récompense que nous avons utilisées sont volontairement peu informatives, le robot n'étant positivement récompensé que lorsqu'il parvient à un état but. A l'inverse, les fonctions de récompense des travaux de pointe utilisent une récompense qui est "presque" une solution du problème, ce qui transforme

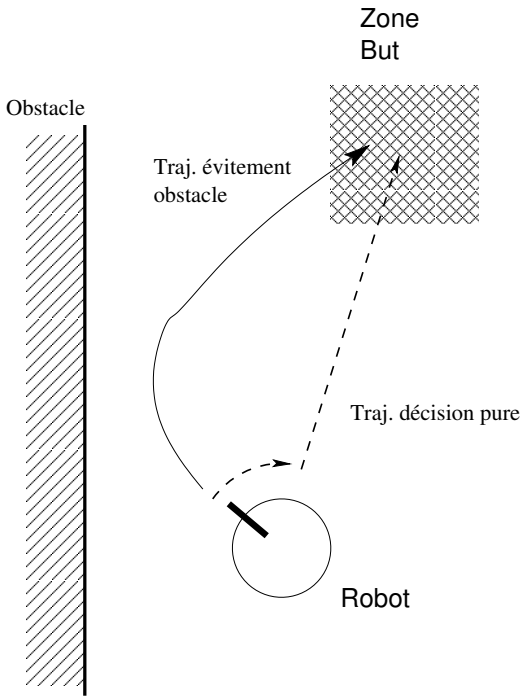


FIGURE 5.7 – **Exploiter l'environnement**. Etant donnée la localisation du robot, on pourrait penser que la décision prise pour rejoindre la zone de but serait d'abord de tourner sur place vers la droite puis d'avancer tout droit (trajectoire en pointillés). Mais le robot "préfère" exploiter son environnement en décidant d'avancer en laissant sa capacité d'éviter les obstacles le rediriger... vers son but (trajectoire en lignes pleines). Cette décision est en fait plus économe – l'agent n'a qu'une action à effectuer au lieu de deux – et, surtout, plus robuste étant donné l'incertitude entourant l'orientation réelle du robot.

presque les problèmes en problèmes d'apprentissage supervisé. Par exemple, dans (Vlassis et al., 2009), la récompense pour qu'un robot pendulaire apprenne à se tenir immobile et vertical est de la forme

$$r(t) = \frac{1}{2}\exp(-x_1^2(t)) + \frac{1}{2}\exp(-x_3^2(t))$$

où x_1 est l'angle entre la verticale et le robot et x_3 la vitesse du robot.

De plus, la finalité de nos travaux n'est pas la même que celles des travaux les plus efficaces et les plus avancés en apprentissage par renforcement pour la robotique. Alors que les travaux précédents, comme de nombreux travaux en robotique évolutionnaire, robotique probabiliste ou même en robotique d'inspiration biologique ont comme finalité de concevoir plus aisément des robots plus performants (voir (Siciliano and Khatib, 2008) pour une présentation plus détaillée de ces domaines), mes travaux se situent dans un cadre de sciences cognitives. Mon but n'est pas seulement de trouver des méthodes de contrôle ou de génération de contrôleurs les plus efficaces mais aussi et surtout d'essayer de mieux comprendre les mécanismes de l'intelligence. A ce titre, ils sont plus proches, dans leurs finalités, des travaux menés par l'équipe de Gerald Edelman (Almassy et al., 1998; Krichmar et al., 2005; Fleischer et al., 2007) ou de (Oudeyer et al., 2007). A travers ses robots "Darwin" qu'il dit "*Brain-Based*", Edelman et son équipe s'intéressent à la motivation dans l'apprentissage de comportement en s'appuyant sur une architecture connexioniste, dans un cadre global qui est en fait celui de l'apprentissage par renforcement, et ce dans le but de mieux comprendre le rôle des différentes aires corticales. Oudeyer se place aussi dans le cadre de l'apprentissage par renforcement pour explorer le concepts de "robot curieux". Bien que les perceptions et les actions du robot soient simplifiées à l'extrême, le but principal d'Oudeyer est d'explorer l'optimisation du taux de renforcement pour que le robot soit toujours à la recherche de la tâche lui permettant d'apprendre au mieux en évitant les tâches trop "faciles" – gain faible en récompense – et les tâches trop "complexes" – gain en récompense négatif car la tâche apparaît comme complètement aléatoire. Dans les deux cas, l'objectif est d'essayer de comprendre des mécanismes fondamentaux de l'intelligence.

Dans une thématique orientée sciences cognitives, une première question qui se pose à l'issue des travaux préliminaires que nous avons effectués sur l'utilisation de l'apprentissage par renforcement dans un cadre robotique est de savoir ce que je veux apporter, ou plutôt, ce que je peux apporter dans ce domaine. Bien que les travaux effectués dans cette optique soient plus rares, ils n'en sont pas moins plus développés et plus avancés. Par contre, ils ont souvent une coloration connexioniste ou neuro-computationnelle et les travaux y mêlant l'apprentissage par renforcement sont plus rares. Il me semble donc que c'est dans cette direction que nous pouvons apporter une contribution.

Dès lors, des questions plus spécifiques mais encore largement ouvertes sont à prendre en compte au vue de nos travaux robotiques. Ainsi, nous avons vu l'importance que l'on doit donner aux **capacités perceptives et motrices du robot**. Il faut trouver un compromis entre la qualité de ces fonctionnalités tout en veillant à ce qu'elles ne soient pas trop spécifiques pour ne pas interférer avec l'objet d'étude principal (par exemple le mécanisme d'apprentissage). Ce compromis touche aussi à la question plus fondamentale de savoir quels sont les mécanismes qui doivent être "innés" pour qu'un robot puisse développer son intelligence.

De plus, nos travaux montrent qu'il faut prêter une grande attention au concept d'"**agent**

complet" par opposition à l'étude d'une fonctionnalité isolée d'un agent. Nous avons vu que le fait de considérer un robot entier, avec donc plusieurs fonctionnalités, permettait de rendre la tâche d'apprentissage plus facile, l'interaction des différentes fonctionnalités compensant les éventuelles faiblesses d'une fonctionnalité. Ainsi, lors de l'apprentissage indirect décrit en section 5.3, l'évitemment d'obstacle peut être exploité pour pallier l'imprécision de capteurs. Considérer le robot comme un tout est aussi au cœur de l'*embodiment*. Par contre, il est clair que considérer le robot dans son ensemble complique considérablement l'étude. C'est en ce sens que, là aussi, un compromis doit être trouvé, et que les motivations derrière ce compromis doivent être explicitées.

Enfin, se pose aussi la question de la forme et de l'origine du ou des **signaux de renforcement** utilisés par le robot. Souvent négligée, la question de la forme du signal de renforcement est importante car entre une récompense concentrée sur quelques états ou un signal de récompense donnant une sorte de distance à l'état but, de nombreuses possibilités existent qui ne sont pas sans influence sur de nombreux aspects de la tâche d'apprentissage (forme de la solution, complexité de l'apprentissage, modélisation, *etc*). Plus encore, c'est la question de l'origine du signal de renforcement qui me paraît marquante. Est-ce un signal externe ou interne ? Comment est-il généré ? Le robot est-il capable de définir par lui-même de nouveaux signaux de renforcement ? Comment ? Quel rapport avec la "motivation", les "émotions" ? Autant de questions qui dépassent parfois le cadre de l'apprentissage ou de la robotique mais qu'il me semble important de prendre en compte quand on se place sous l'angle des sciences cognitives.

Ces questions, et l'importance qu'elles vont prendre dans mes travaux futurs, est développée dans la partie suivante qui explicite les questions scientifiques que je compte explorer au cours de mes recherches à venir.

6

Synthèse du projet de recherche

6.1 Avant propos

Nous en arrivons maintenant à ce qu'on appelle couramment un *projet de recherche*. En guise d'avant-propos, je tiens à signaler que si certains aspects du terme *projet* me plaisent – notamment le fait qu'un projet est par nature quelque chose qui est amené à être révisé lors de son avancement – certaines de ses caractéristiques me gênent un peu. En particulier, dans le système de recherche actuel la notion de projet est de plus en plus associée avec la notion de cahier des charges, d'échéancier, d'évaluation du risque *a priori*, de budget, *etc.* Or, en ce qui me concerne, je trouve qu'il est difficile de faire rentrer la recherche dans ce cadre un peu trop strict, "comptable" et où le problème implicite est la "productivité". Le risque est inhérent à la recherche. Un échéancier est souvent utopique car les résultats ne se commandent pas. Il est difficile de délimiter exactement un problème, car cela voudrait souvent dire qu'il est déjà résolu, et donc hors du domaine de la recherche.

Pour moi, un "projet" de recherche est plus affaire de questions que de réponses. L'essentiel du projet sera alors d'explorer et de traiter ces questions de manière scientifique, en s'attachant à bien expliciter le cadre de la recherche, à définir les hypothèses que l'on cherche à vérifier et à solidement étayer les conclusions avancées. On trouvera donc ici des questions et des voies que je compte explorer en espérant qu'elles me permettront de commencer à répondre à ces questions...

6.2 Problématique générale

La question centrale à long terme de mon projet de recherche est la suivante :

"*Comment une entité artificielle élabore-t-elle ses représentations de soi et de son environnement ?*"

Pour être un peu plus explicite, cette question renvoie à la question de savoir comment se constitue et évolue "l'état interne" d'un agent à partir de ses sensations et perceptions immédiates. Des notions comme le "contexte" ou la "situation présente" sont aussi à relier à ce problème. C'est en fonction de ses représentations que l'agent artificiel va choisir et mettre en œuvre ses actions. Bien qu'elle ait longtemps été occultée par l'intelligence artificielle classique qui n'y voyait aucune difficulté et qui théorisait que l'élaboration des représentations, forcément symboliques, n'était qu'une question technique de traitement du signal, cette problématique n'est toujours pas résolue. Cette question est d'ailleurs cruciale à la compréhension des mécanismes de l'intelligence, en particulier quand on se place dans le courant de pensée de la cognition incarnée ("*embodied cognition*" ou, par abus de langage, "*embodiment*") – cf. section 2.1.

A l'image de Dreyfus, qui voit dans l'apprentissage par renforcement la démarche la plus avancée sur la voie de l'*embodiment*[18] (Dreyfus, 1993), je compte explorer la question de l'élaboration de représentations par le biais de l'apprentissage par renforcement.

La principale raison qui motive ce choix est que l'on peut voir l'apprentissage par renforcement comme une théorie synthétique de l'*embodiment*, c'est-à-dire un cadre théorique indiquant comment construire un agent intelligent en s'appuyant sur les concepts de l'*embodiment*. Ainsi que je le présente en section 2.2, l'apprentissage par renforcement explique comment un agent peut apprendre de manière autonome des comportements "intelligents" en cherchant à optimiser un critère de performance dépendant d'un signal de récompense par le biais d'interactions répétées avec son environnement. S'appliquant à des espaces d'états et d'actions continus, les algorithmes sont compatibles avec l'approche numérique et non-symbolique exigée par l'*embodiment*. Bien que la preuve par l'exemple ne soit pas en soi une preuve, c'est à l'heure actuelle la seule solution envisagée et envisageable pour "prouver" l'*embodiment*.

Reformulée dans le cadre de l'apprentissage par renforcement, la question de l'élaboration de représentations par un agent devient celle de l'apprentissage par renforcement pour un agent confronté à une *tâche non-markovienne*. Ce cadre est celui d'un agent dont les perceptions instantanées et courantes ne lui apportent pas assez d'information pour pouvoir prédire les conséquences de ses actions, comme je l'explique plus longuement aux sections 2.3.2 et 2.4.3. Pour résoudre ce problème de manière autonome, l'agent est alors obligé de se construire une représentation interne adaptée au problème, c'est-à-dire ni trop pauvre – l'agent serait alors incapable d'apprendre, ni trop riche – l'agent serait alors incapable de gérer et de manipuler cette représentation.

Des représentations mathématiques permettant de résoudre des problèmes non-markoviens existent. C'est notamment le cas des *belief states*[19] et des *PSR*[20]. Ces représentations "comptables" et "désincarnées" s'appuient sur un traitement purement statistique de l'information. Elles sont inadaptées au problème de l'apprentissage, lourdes à mettre en œuvre et limitées à des problèmes somme toute assez simples. En ce qui concerne le domaine de l'apprentissage par renforcement, les problèmes non-markoviens sont en fait encore largement ouverts.

18. Cette conclusion de Dreyfus s'appuie sur un vue dépassée du connexionisme, notamment par le fait qu'il n'envisage que l'apprentissage supervisé dans le cadre des réseaux de neurones, mais elle souligne néanmoins les rapports étroits entre les deux cadres.
19. Voir section 3.1.
20. De l'anglais "*Predictive State Representation*", vues en section 3.1.

Mes constatations à l'issue de plusieurs travaux de recherche touchant spécifiquement à cette problématique d'apprentissage par renforcement dans un cadre non markovien (voir la section 3 et notamment ses conclusions en section 3.6) tendent à montrer qu'une approche "désincarnée" ne parviendra pas à apporter des réponses satisfaisantes.

En fait, l'argumentation sur laquelle on retombe est assez similaire à celle qui a motivé le courant de pensée de l'*embodiment*. On retrouve les problèmes "d'être en situation" et du "*frame problem*" évoqués précédemment. Se construire une représentation adéquate de son environnement revient à détecter, parmi une myriade de possibilités, les causalités et régularités sensori-motrices qui sont *pertinentes*, c'est-à-dire qui permettent à l'agent de résoudre ses problèmes. La conclusion à laquelle on arrive alors est que cette détection est impossible à mettre en œuvre car, pour résumer, il y a trop de cas à examiner et le seul moyen de se tirer d'affaire serait d'avoir déjà "reconnu" le contexte pour pouvoir focaliser la recherche ce qui demanderait d'avoir auparavant détecté quelles étaient les causalités pertinentes. Il y a la un cycle infernal dont on ne sait comment sortir.

Le courant de pensée de l'*embodiment* propose de sortir de cette boucle en ancrant l'agent artificiel dans son monde, principalement par le biais de son corps. Parmi les nombreux principes suggérés par l'*embodiment*, je compte m'appuyer principalement sur une approche incrémentale, holistique et motivationnelle de l'apprentissage par renforcement.

6.3 Apprentissage par renforcement incrémental, holistique et motivationnel

En faisant apprendre à des robots des tâches non-markoviennes, ces derniers devraient se doter de représentations pertinentes et adaptées de leur environnement. Pour ce faire, je compte mettre en place des apprentissages qui soient incrémentaux, holistiques et motivationnels, ce que je détaille maintenant.

Une approche incrémentale de l'apprentissage. J'ai déjà commencé à explorer une facette de cette approche incrémentale au cours de travaux effectués avec Olivier Buffet sur des systèmes mono- et multi-agents (voir sections 3.5 et 4.4). L'idée alors mise en avant était de placer les agents face à des situations de plus en plus complexes pour qu'ils apprennent et réutilisent des comportements afin de construire des comportements de plus en plus complexes et "intelligents". Je compte reprendre et continuer ces travaux en y ajoutant de plus d'autres dimensions pour lesquelles il est possible de progresser de manière incrémentale. Je pense en particulier aux dimensions sensorielles et motrices où, par un biais logiciel ou matériel, il est possible de doter les agents de capacités évoluant avec le temps ou avec leur habileté à résoudre une tâche.

Il est en effet postulé que les contraintes et limitations morphologiques chez les êtres vivants leur permettent en fait d'augmenter leur adaptativité lorsqu'ils se développent. Au début, ils ne sont pas noyés sous une masse d'information qui les empêcherait d'apprendre puis leurs sens et capacités motrices s'aiguisent au fur et à mesure qu'ils acquièrent les capacités cognitives pour

les gérer (Turkewitz and Kenny, 1982; Hendriks-Jensen, 1996).

Une approche holistique de l'apprentissage. Parmi les intérêts du travail effectué avec Nicolas Beaufort sur une approche indirecte de l'apprentissage par renforcement sur un robot mobile présenté en section 5.3, le fait d'avoir considéré l'apprentissage comme une des fonctions d'un robot, n'étant alors qu'une partie d'un tout plus complexe, a rendu le robot plus performant. Du fait de sa capacité "innée" à éviter les obstacles, le robot est devenu plus robuste et a pu apprendre à naviguer malgré un environnement rendu difficile à cause, notamment, de perceptions limitées et bruitées. Comme les autres fonctions cognitives, l'apprentissage doit être pensé non seulement en fonction de la tâche à apprendre mais aussi en fonction de l'agent, de sa morphologie et de ses autres fonctions cognitives. A mon avis, c'est d'autant plus vrai dans le cas de l'apprentissage d'une tâche non-markovienne car, ainsi que nous l'avons observés dans nos différents travaux, ce que l'on peut apprendre dépend encore plus étroitement du comportement global de l'agent. Formellement, il a été montré que la politique "adaptée" apprise dans un POMDP est fonction de la stratégie d'exploration de l'agent (voir section 3.2).

Par contre, considérer l'agent dans sa globalité n'est pas sans difficultés supplémentaires. Il est plus délicat de considérer un ensemble de parties que chacune de ces parties indépendamment et c'est pourtant ce que sous-tend cette approche "holistique". Comme de plus les différentes fonctionnalités cognitives de l'agent ne sont pas forcément organisées en modules indépendants, cette approche peut vraiment compliquer la construction, la compréhension, la mise au point et la validation des différentes fonctionnalités de l'agent.

Une approche motivationnelle de l'apprentissage. Les motivations jouent un rôle important dans nos capacités cognitives. Cet aspect, bien que primordial, n'est pas assez pris en compte dans le formalisme de l'apprentissage par renforcement. L'origine et la nature du signal de récompense n'est pas du tout important, il suffit juste de spécifier si la récompense est liée à tel ou tel état, ou couple d'état et d'action, voire à telle transition. Dans le cadre de l'*embodiment*, la nature et l'origine de ce signal est primordiale. Dans tous les cas, le signal est un signal interne à l'agent, signal qu'il a lui-même généré. Si des conditions particulières de l'environnement peuvent être la cause du signal (comme par exemple un choc électrique) le "véritable" signal (par exemple la douleur) est interne. L'attribution d'une récompense, au sens de l'apprentissage par renforcement, à tel ou tel signal, est un mécanisme qui est mal connu et très peu étudié dans la communauté de l'intelligence artificielle. Et doter un robot d'un mécanisme semblable est encore une activité qui est, au mieux, très expérimentale. Qu'est-ce que la "douleur", la "faim" ou "la fatigue" pour un robot ? Comment le lui faire ressentir ? Bien que cette approche apporte plus de questions que de réponses, il existe plusieurs travaux qui proposent des approches intéressantes dont il est possible de s'inspirer, par exemple (Alexander and Sporns, 2004; Sporns and Alexander, 2002; Oudeyer et al., 2007).

6.4 Directions de recherche

Les trois approches détaillées ci-dessus, et l'*embodiment* en général, nécessitent donc des agents "incarnés". C'est pourquoi mes recherches s'orientent résolument vers l'utilisation de robots, le but étant moins de faire progresser la robotique en tant que telle que d'incarner les agents artificiels dans une réalité riche et complexe. C'est à cette condition que des interactions intéressantes peuvent se développer entre l'agent et son environnement, interactions qui sont à la base de la cognition. Et c'est aussi à cette condition que je pourrai contribuer à l'*embodiment*, j'entends par là contribuer à l'exploration et la compréhension de mécanismes permettant la cognition humaine ou artificielle.

Pour explorer et mettre en place un apprentissage robotique s'appuyant sur les trois principes vus auparavant, je compte développer mon projet de recherche selon trois voies interdépendantes.

Axe de la robotique cognitive développementale. Le domaine de la robotique développementale s'est développé il y a moins d'une dizaine d'années au croisement de l'*embodiment*, des sciences du développement et de la robotique. Une partie des travaux effectués avec Olivier Buffet en 2001-2003 peuvent entrer dans le cadre de ce courant de pensée bien que n'ayant pas été portés sur un robot. Dans un premier temps, je projette de combiner ces travaux avec mes travaux plus récents en robotique (voir section 5) pour y explorer les apports des concepts de la robotique développementale pour l'apprentissage de tâches non-markoviennes.

La robotique développementale a en effet formalisé plusieurs concepts fondamentaux, parmi ces derniers je compte m'appuyer sur la plupart de ceux listés et détaillés dans (Lungarella et al., 2003). Par ces travaux, j'aspire à analyser comment un robot développe des représentations qui lui sont propres et qui vont lui permettre de résoudre une tâche non markovienne tout en tenant compte des limitations qui ont émergé de nos travaux en robotique. Ainsi, mes recherches se concentreront essentiellement sur les aspects **incrémental** et **holistique** de l'apprentissage au travers des questions suivantes :

- *Quelles capacités innées pour le robot ?* Quelles sont les capacités sensori-motrices, mais aussi cognitives, dont doit disposer le robot pour qu'il puisse apprendre ? Cette question renvoie aussi à la question de savoir à quel niveau d'abstraction l'apprentissage par renforcement est le plus utile, comme par exemple pour apprendre à avancer (très bas niveau) ou apprendre à naviguer (fonction de plus haut-niveau). On devra aussi se poser la question de la redondance des modalités perceptives du robot et de leur influence ou intérêt sur la possibilité d'élaborer des représentations ;
- *Sous quelle forme fondamentale manipuler les données sensori-motrices ?* Cette question renvoie en partie à la précédente car les données manipulées dépendent bien évidemment des données sensori-motrices et du niveau d'abstraction auquel se place l'apprentissage. J'en fais un point à part pour insister sur le fait que l'*embodiment* postule que l'agent doit d'abord travailler à un niveau sub-symbolique alors que, dans le cadre de la robotique et de l'apprentissage par renforcement, cette contrainte n'est pas forcément aisée à respecter. Ce point pose aussi la question de la validité d'un mécanisme mis en place en fonction du "degré" de symbolisme utilisé lors de sa mise au point ;
- *Comment utiliser au mieux chaque interaction avec l'environnement ?* Ce point a déjà été abordé lors de nos travaux en robotique. L'idée principale est que les interactions entre un

robot et son environnement sont coûteuses en temps et parfois dangereuses pour l'intégrité du robot. Les méthodes développées doivent donc tirer le maximum de chaque interaction, au risque de faire des approximations trop grossières ou d'amener à des comportements clairement sous-optimaux ;
- *Comment agencer et modifier les tâches et les capacités sensori-motrices du robot pour lui permettre d'apprendre ?* Je ne reviens pas sur ce point qui est central à la notion d'apprentissage incrémental tel que je l'ai présenté précédemment.

La méthodologie que je compte mettre en œuvre dans cet axe de recherche sera principalement expérimentale, s'appuyant sur les plateformes robotiques présentes et à venir du laboratoire. Ainsi, dans un premier temps, je compte explorer les possibilités offertes par le fait de faire évoluer les capacités perceptives et/ou motrices d'un agent au fur et à mesure que son expérience et ses performances cognitives et comportementales augmentent. Cela peut se faire dans des tâches de navigation avec des robots khepera dont, par exemple, les images fournies par une caméra deviendraient de moins en moins floues. De même, en utilisant un robot Nao, nous envisageons des tâches de pointage ou de cordination "bras articulé/vision" où la vision du robot deviendrait de plus en plus détaillée et où le bras à manipuler disposerait de plus en plus de degré de liberté à contrôler.

Axe des neurosciences computationnelles. Dans mon exploration de l'élaboration de représentation, il me semble aussi important de me tourner vers le "vivant", humain ou animal, qui fait indéniablement preuve d'intelligence et qui a su résoudre ce problème. Ainsi, les neurosciences computationnelles sont évidemment une source naturelle d'inspiration mais je pense aussi que c'est un domaine où je peux, très modestement, contribuer. C'est dans l'idée de contribuer aux approches **holistique** et **motivationnelle** de l'apprentissage que je compte mener ces travaux.

Mes intérêts sont multiples dans ce domaine mais mes connaissances et mes acquis encore très limités. De plus, c'est un domaine énorme d'où peuvent émerger un nombre fantastique de projets de recherche. Aussi, pour ne pas s'y perdre, il faut se faire une vision assez précise et délimitée de ce que l'on y cherche et de ce que l'on peut espérer y faire. Pour cela, mon objectif est d'avancer sur cette voie par le biais de collaborations incluant des chercheurs en neurosciences computationnelles qui sont plus à même de faire le pont entre le monde des biologistes et le monde de l'intelligence artificielle. Les diverses collaborations en cours avec l'équipe CORTEX du Loria sont mes premiers pas dans cette voie.

Pour être plus précis, je détaille ci-dessous comment je vois mes intérêts et objectifs dans cet axe de travail. Une large part est accordée à l'acquisition de connaissances qui seront réutilisées comme source d'inspiration, essentiellement en robotique. Mais au cours même de cette aquisition de connaissance je compte aussi contribuer à ce domaine. C'est ce que je fais dès a présent, notamment par le biais du co-encadrement de la thèse d'Elham Ghassemi sur le rôle du cervelet dans la boucle sensori-motrice des saccades occulaires (voir projet MAPS, page 102).
- *Etude des boucles sensori-motrices.* L'objectif est d'extraire les principes fondamentaux en terme d'information véhiculée, de codage et d'interaction au sein des boucles sensori-motrices du vivant. Il y a là une source d'inspiration pour contribuer à construire et définir l'agent robotique dans l'optique de l'apprentissage **holistique** ;
- *Fonction d'apprentissage par renforcement.* Outre ses mécanismes neuronaux, ce sont les

autres fonctionalités cognitives avec lesquelles interagit l'apprentissage qui m'intéresse. Le but est de comprendre ce qu'apportent ces autres fonctionnalités et de s'en inspirer pour apporter à l'agent robotique complet des capacités lui permettant d'apprendre des tâches non-markoviennes ;

– *Apprentissage synaptique neuromodulé.* A niveau des cartes neuromimétiques et de l'apprentissage se déroulant dans les synapses mêmes des neurones, il y a parfois besoin d'un apprentissage "motivé" (Sporns and Alexander, 2002). Comment une notion de récompense ou de motivation est-elle véhiculée, distribuée et utilisée par les différentes synapses ? C'est plus sur cette problématique que je pense pouvoir contribuer, tout en en retirerant des inspirations pour les systèmes multi-agents qui constituent le troisième axe de travail de mon projet.

Comme je le disais, cet axe de recherche s'articulera principalement autour de collaborations, notamment avec des biologistes et des chercheurs en neuro-sciences computationnelles, au moins dans un premier temps. A cet effet, je compte développer les coopérations avec l'équipe Cortex du Loria car nous avons en commun de vouloir comprendre la cognition par une approche intégrative et holistique. Les premières pierres de ces collaborations ont été posées au travers de l'ANR MAPS (voir page 102), du co-encadrement de la thèse d'Elham Ghassemi (voir plus haut). Dans le même ordre d'idée, ces collaboratins concernent aussi le projet *Adaptation & Action* avec Alain Marchand du Centre de Neurosciences Intégratives et Cognitives (CNIC) de Bordeaux avec qui nous essayons de comprendre quels sont les mécanismes d'apprentissage par renforcement utilisé par les rats dans leur prise de décision (voir page 102). Avec Yann Boniface, de Cortex et Emmanuel Daucé de l'UMR Mouvement et Perception de Marseille, nous avons aussi comme projet d'adapter des règles d'apprentissage par renforcement à des modèles connexionistes pour motiver et neuromoduler l'apprentissage.

Axe de l'apprentissage auto-organisé dans les systèmes multi-agents. Comme je le vois, cet axe de recherche sur les systèmes multi-agents aura curieusement plus de liens avec l'axe précédent sur la neurocomputation qu'avec mes travaux antérieurs sur l'apprentissage par renforcement dans les systèmes multi-agents. De mes précédents travaux, je tire la conclusion que l'apprentissage par renforcement multi-agent, bien que fort intéressant et confronté à de nombreux problémes ouverts, ne me sera pas d'une grande aide pour comprendre comment un agent peut élaborer des représentations. Il serait certes intéressant de travailler sur les représentations que peut se faire un agent des autres agents, mais pour ma part, je pense que les réflexions avec un seul agent ne sont pas assez avancées pour passer tout de suite au cas multi-agent. Je parle du point de vue de la recherche sur les mécanismes cognitifs de l'élaboration de représentations, pas d'un point de vue plus général. Ainsi, utiliser l'apprentissage par renforcement pour faire se coordonner des robots est certes intéressant, mais nécessite de se placer à un tel niveau d'abstraction que, de mon point de vue, les mécanismes fondamentaux de la cognition sont escamotés, faisant ainsi réapparaître l'homoncule de la section 2.1.

Par contre, la notion d'émergence ou d'auto-organisation, qui est au centre de systèmes composés de nombreux agents — on parle aussi de systèmes complexes ou d'intelligence en essaim (Beni and Payton, 2005) ; est aussi cruciale pour l'*embodiment*. On y postule en effet que les représentations doivent *émerger* des interactions de l'agent avec son environnement, principalement par *auto-organisation* de ses boucles sensori-motrices. De plus, il existe une certaine similitude entre un large groupe d'agents apprenant par renforcement et une population de neu-

rones dont l'apprentissage est neuromodulé. Les questions précédentes sur comment le signal est "véhiculé, distribué et utilisé" s'appliquent aux agents.

Je compte aborder cet axe selon plusieurs points qui me paraissent de difficultés croissantes. Ces points seraient les suivants :

- *Apprentissage distribué.* La question principale est de savoir comment *un* signal de renforcement peut être utile à un ensemble d'agents. En particulier, il faut savoir comment le signal est véhiculé, distribué et utilisé mais aussi quelle est son origine, comment il est synthétisé. En ce sens, les réponses à ce niveau peuvent clairement contribuer aux approches **holistiques** et **motivationnelles** de l'apprentissage ;
- *Auto-organisation et apprentissage.* La question principale de cet axe de travail sera d'explorer les effets résultants de l'apprentissage individuel des agents dans le cadre de l'intelligence en essaim. Est-ce que l'apprentissage individuel peut accélérer l'auto-organisation, augmenter la robustesse ? Est-ce un moyen de "diriger" un peu l'auto-organisation ? A ma connaissance, ces questions n'ont pas été étudiées et restent donc encore largement ouvertes ;
- *Caractériser l'auto-organisation.* Une des questions centrales des systèmes complexes est de détecter et de mesurer l'auto-organisation. Cette problématique peut avoir des liens avec le problème de la détection de régularités sensori-motrices par et pour les agents, problème fortement connecté à l'élaboration de représentations. En ce qui me concerne, cette question me paraît encore floue et mal définie, et il ne me semble pas raisonnable de m'y attaquer dès maintenant, sans une réflexion supplémentaires approfondie. Mais je sens qu'elle ne sera pas sans retombées en ce qui concerne l'élaboration de représentations pertinentes pour des agents.

Là aussi, la méthodologie que je vais suivre sera principalement expérimentale et devra s'articuler en collaboration avec une partie de l'équipe Maia qui, sous la houlette de Vincent Chevrier et Nazim Fatès, s'intéresse de plus en plus à l'auto-organisation et la robustesse des systèmes multi-agents. Comme je l'envisage, les premiers travaux pourraient concerner des modèles commes celui des araignées sociales (Bourjot et al., 2003) ou du projet Amybia (Fatès, 2010) dans lequels les agents seraient individuellement capables de s'adapter *en-ligne* avec un feedback extérieur – un signal de renforcement – calculé globalement. Les premières questions à étudier seraient alors de savoir en quoi cette adaptation individuelle perturbe ou améliore le processus d'auto-organisation. Ce mécanisme a-t-il une influence sur la robustesse de ces systèmes ? Comment le signal global est-il véhiculé aux agents ? *Etc.* Parallèlement, il sera intéressant de caractériser plus quantitavement ces comportements et ces influences et d'essayer d'en retirer des informations pour mieux comprendre ou analyser les phénomènes d'auto-organisation et d'émergence.

6.5 En conclusion

En résumé, je cherche à comprendre certains mécanismes de la cognition en essayant de construire des robots qui apprennent des représentations sensori-motrices de leur monde. Pour cela, je vais m'intéresser à des mécanismes d'apprentissage par renforcement holistiques, incrémentaux et motivationels de tâches non-markoviennes en m'appuyant sur des travaux concernant la robotique cognitive développementale, les neurosciences computationnelles et l'auto-organisation dans les systèmes multi-agents. J'espère ainsi pouvoir contribuer à la compréhension de concepts

clefs des sciences cognitives comme l'ancrage des symboles, l'équilibre entre l'inné et l'acquis et, sans doute très modestement à très long terme, à l'élaboration d'une conscience de soi.

Bibliographie

Aberdeen, D. and Buffet, O. (2007). Temporal probabilistic planning with policy-gradients. In *Proceedings of the Seventeenth International Conference on Automated Planning and Scheduling (ICAPS'07)*. 20

Aberdeen, D., Buffet, O., and Thomas, O. (2007). Policy-gradients for PSRs and POMDPs. In *Proc. of the Eleventh Int. Conf. on Artificial Intelligence and Statistics (AISTATS'07)*. 31

Alexander, W. and Sporns, O. (2004). Interactions of environment, behavior and synaptic patterns in a neuro-robotic model. In *From Animals to Animats 8 (SAB'04)*, pages 13–22. 74

Almassy, N., Edelman, G. M., and Sporns, O. (1998). Behavioral constraints in the development of neuronal properties : a cortical model embedded in a real-world device. *Cereb. Cortex*, 8(4) :346–361. 69

Amir, E. (2005). Learning partially observable deterministic action models. In *Proc. of the Nineteenth Int. Joint Conf. on Artificial Intelligence (IJCAI'05)*, pages 1433–1439. 33

Aras, R. (2008). *Mathematical Programming Methods for Decentralized POMDPs*. PhD thesis, Université Henri Poincaré - Nancy I. 46

Aras, R. and Dutech, A. (2009). An investigation into Mathematical Programming for Finite Horizon Decentralized POMDPs. Research Report RR-7066, INRIA. 24, 46, 47

Aras, R., Dutech, A., and Charpillet, F. (2004). Cooperation through communication in decentralized Markov games. In *Proc. of the Int. Conf. on Advances in Intelligent Systems - Theory and Applications (AISTA)*, Luxembourg-Kirchberg, Luxembourg. 24, 50

Aras, R., Dutech, A., and Charpillet, F. (2005). Cooperation in stochastic games through communication. In *Proc. of the fourth Int. Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS'05)*, Utrech, Nederlands. 24

Aras, R., Dutech, A., and Charpillet, F. (2006). Efficient learning in games. In *Actes de la huitième Conf. Francophone sur l'Apprentissage (CAp'06)*, Trégastel, France. 24, 51

Aras, R., Dutech, A., and Charpillet, F. (2007a). Mixed integer linear programming for exact finite-horizon planning in decentralized POMDPs. In *Proc. of the Int. Conf. on Automated Planning & Scheduling (ICAPS'07)*. 24, 46

Aras, R., Dutech, A., and Charpillet, F. (2007b). Une méthode de programmation linéaire mixte pour les POMDP décentralisés à horizon fini. In *Actes des Journées Françaises de Planification, Décision, Apprentissage (JFPDA)*. 24, 46

Asada, M., Noda, S., Tawaratsumida, S., and Hosoda, K. (1996). Purposive behavior acquisition for a real robot by vision-based reinforcement learning. *Machine Learning*, 23 :279–303. 52

Aström, K. (1965). Optimal control of Markov decision processes with incomplete state estimation. *Journal of Mathematical Analysis and Applications*, 10 :174–205. 29

Atkeson, C., Moore, A., and Schaal, S. (1997). Locally weighted learning. *AI review*, 11 :11–73. 61

Avila-Garcia, O. and Canamero, L. (2004). Using hormonal feedback to modulate action selection in a competitive scenario. In *From Animals to Animats. Proc. of the 8th Int. Conf. of Simulation of Adaptive Behavior (SAB'04)*. 37

Barto, A., Bratke, S., and Singh, S. (1995). Learning to act using real-time dynamic programming. *Artificial Intelligence*, 72 :81–138. 19, 66

Baxter, J. and Bartlett, P. (2000). Reinforcement learning in POMDP's via direct gradient ascent. In *Proc. 17th International Conf. on Machine Learning (ICML'00)*. 30, 31

Baxter, J. and Bartlett, P. (2001). Infinite-horizon policy-gradient estimation. *Journal of Artificial Intelligence Research*, 15 :319–350. 17

Baxter, J., Bartlett, P., and Weaver, L. (2001). Experiments with infinite-horizon, policy-gradient estimation. *Journal of Artificial Intelligence Research*, 15 :351–381. 17, 31

Beaufort, N. (2009). Apprentissage optimiste et planification partielle pour un robot mobile. Master's thesis, Université Henri Poincaré, Nancy I. 25, 64

Bellman, R. (1957). *Dynamic programming*. Princeton University Press, Princeton, New-Jersey. 20

Beni, G. and Payton, D., editors (2005). *Swarm Robotics*, volume 3342 of *Lecture Notes in Computer Science*, pages 1–9. Springer Berlin Heidelberg. 77

Bernstein, D., Zilberstein, S., and Immerman, N. (2000). The complexity of decentralized control of Markov decision processes. In *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence, Stanford, California*. 42, 44

Bertsekas, D. and Tsitsiklis, J. (1996). *Neuro-dynamic programming*. Athena Scientific, Belmont, MA. 13, 18

Bonabeau, E., Dorigo, M., and Theraulaz, G. (1999). *Swarm Intelligence : From Natural to Artificial Systems*. Oxford University Press, USA. 39

Bourjot, C., Chevrier, V., and Thomas, V. (2003). A new swarm mechanism based on social spiders colonies : from web weaving to region detection. *Web Intelligence and Agent Systems : An International Journal*, 1(1) :47–64. 78

Boutilier, C. (1996). Planning, learning and coordination in multiagent decision processes. In *Proceedings of the 6th Conference on Theoretical Aspects of Rationality and Knowledge (TARK '96), De Zeeuwse Stromen, Nederlands.* 40

Boutilier, C. (1999). Sequential optimality and coordination in multiagent systems. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI '99), Stockholm, Sweden.* 18

Boutilier, C., Dearden, R., and Goldszmidt, M. (1995). Exploiting structure in policy construction. In *Proc. of the 14th Int. Joint Conf. on Artificial Intelligence (IJCAI'95).* 18

Bowling, M. (2004). Convergence and no-regret in multiagent learning. In *Advances in Neural Information Processing Systems 17 (NIPS'04)*, pages 209–216. 49, 51

Bowling, M. and Veloso, M. (2002). Multiagent learning using a variable learning rate. *Artificial Intelligence*, 136 :215–250. 49

Boyan, J. (1999). Least-squares temporal difference learning. In *Proc. of the 16th Int. Conf. on Machine Learning (ICML'99)*, pages 49–56. 18

Bradtke, S. and Barto, A. (1996). Linear least-squares algorithms for temporal difference learning. 22 :33–57. 18

Braitenberg, V. (1984). *Vehicules : Experiments in synthetic psychology.* The MIT Press, Cambridge. 9, 64

Brooks, R. (1986). Asynchronous distributed control system for a mobile robot. In *SPIE Conference on Mobile Robots*, pages 77–84. 67

Brooks, R. (1991). Intelligence without reason. In *Proceedings of the Int. Joint Conf. on Artificial Intelligence.* 3, 7

Brooks, R. (1999). *Cambrian Intelligence : the early history of the New AI.* The MIT Press, Cambridge. 7

Browne, W., Kawamura, K., Krichmar, J., Harwin, W., and Wagatsuma, H. (2009). Cognitive robotics : new insights into robot and human intelligence by reverse engineering brain functions [from the guest editors]. *Robotics & Automation Magazine, IEEE*, 16(3) :17–18. 5

Buffet, O. (2003). *Une double approche modulaire de l'apprentissage par renforcement pour des agents intelligents adaptatifs.* PhD thesis, Université Henri Poincaré, Nancy 1. 35

Buffet, O., Dutech, A., and Charpillet, F. (2001). Incremental reinforcement learning for designing multi-agent systems. In *Fifth International Conference on Autonomous Agents, Agents'01 (poster session).* 24, 52

Buffet, O., Dutech, A., and Charpillet, F. (2002a). Adaptive combination of behaviors in an agent. In *European Conference on Artificial Intelligence (ECAI'02).* 23

Buffet, O., Dutech, A., and Charpillet, F. (2002b). Learning to weigh basic behaviors in scalable agents. In *Proc. of the Int. Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS'02)*, Bologna, Italy. 23, 36

Buffet, O., Dutech, A., and Charpillet, F. (2003). Automatic generation of an agent's basic behaviors. In *Proc. of the Int. Conf. on Autonomous Agents and Multi-Agent System (AAMAS'03)*, Camberra, Australia. 23

Buffet, O., Dutech, A., and Charpillet, F. (2004). Self-growth of basic behaviors in an action selection based agent. In *From Animals to Animats. Proc. of Int. Conf. on Simulation of Adaptive Behavior (SAB'04)*, Los Angeles, USA. 23, 36

Buffet, O., Dutech, A., and Charpillet, F. (2005). Développement autonome des comportements de base d'un agent. *Revue d'Intelligence Artificielle (RIA)*, 19(4-5) :603–632. 23

Buffet, O., Dutech, A., and Charpillet, F. (2006). Etude de différentes combinaisons de comportements adaptatives. *Revue d'Intelligence Artificielle (RIA)*, 20(2-3) :311–344. 23, 36

Buffet, O., Dutech, A., and Charpillet, F. (2007). Shaping multi-agent systems with gradient reinforcement learning. *Autonomous Agent and Multi-Agent System Journal (AAMASJ)*, 15(2) :197–220. 24, 52, 53

Caruana, R. (1997). *Learning to Learn*, chapter Multitask Learning. Kluwer Academic Publishers. 16

Caruana, R., Baxter, J., Mitchell, T., Pratt, L., Silver, D., and Thrun, S. (1995). *Learning to learn : knowledge consolidation and transfer in inductive systems.* A NIPS*95 Post-Conference Workshop. http ://www.cs.cmu.edu/afs/cs.cmu.edu/user/caruana/pub/transfer.html. 16

Cassandra, A. (1998). *Exact and Approximate Algorithms for Partially Observable Markov Decision Processes.* PhD thesis, Brown University, Department of Computer Science, Providence, RI. 27

Cassandra, A., Kaelbling, L., and Littman, M. (1994). Acting optimally in partially observable stochastic domains. In *Proc. of the 12th Nat. Conf. on Artificial Intelligence (AAAI).* 30

Chadès, I. (2003). *Planification distribuée dans les systèmes multi-agents à l'aide de processus décisionnels de Markov.* PhD thesis, Université Henri Poincaré - Nancy 1 (UHP). 43

Chadès, I. (2006). Algorithmes de co-évolution simultanée pour la résolution approchée de PDM multi-agent. *Revue d'Intelligence Artificielle (RIA)*, 20 :345–382. 43

Chrisman, L. (1992). Reinforcement learning with perceptual aliasing : The perceptual distinctions approach. In *Proc. of the Tenth National Conf. on Artificial Intelligence (AAAI'92)*, pages 183–188. 32

Cohn, D., Ghahramani, Z., and Jordan, M. (1996). Active learning with statistical models. *Journal of Artificial Intelligence Research*, 4 :129–145. 61

Coulom, R. (2002). *Reinforcement learning using Neural Networks, with Applications to Motor Control.* PhD thesis, Institut National Polytechnique de Grenoble. 18

Coulom, R. (2007). Computing Elo ratings of move patterns in the game of Go. In *Computer Games Workshop*, Amsterdam, The Netherlands. 19

Crites, R. and Barto, A. (1996). Improving elevator performance using reinforcement learning. In *Advances in Neural Information Processing Systems 8 (NIPS).* 12

Cushing, W., Kambhampati, S., Mausam, and Weld, D. S. (2007). When is temporal planning really temporal ? In *Proc. of the 20th International Joint Conference on Artificial Intelligence, (IJCAI'07)*, pages 1852–1859, Hyderabad, India. 20

Dantzig, G. (1991). *K. Lenstra, A. Rinnooy, K. Schrijver and A. Schrijver (eds.) History of mathematical programming*, chapter Linear programming. The story about how it began : some legends, a little about its historical significance, and comments about where its many mathematical programming extensions may be headed, pages 19–31. North-Holland, New York. 24

Dantzig, G. (1998). *Linear Programming and Extensions*. Princeton University Press. 24

Daw, N. D., Niv, Y., and Dayan, P. (2005). Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control. *Nature Neuroscience*, 8(12) :1704–1711. 64

Deflandre, L. (2006). Agent épervier. Master's thesis, Master Informatique, spécialité PRIM - UHP Nancy1. 25, 57

Degris, T. (2007). *Apprentissage par Renforcement dans les processus de décision markoviens factorisés*. PhD thesis, Université Pierre et Marie Curie - Paris 6. 18

Dietterich, T. (2000). Hierarchical reinforcement learning with the MAXQ value function decomposition. *Journal of Artificial Intelligence Research (JAIR)*, 13 :227–303. 18, 37

Digney, B. (1998). Learning hierarchical control structures for multiple tasks and changing environments. In *From Animals to Animats. Proc. of the Fifth Conf. on the Simulation of Adaptive Behavior : SAB 98*. 37

Dreyfus, H. (1993). *What computers still can't do*. The MIT Press, Cambridge, MA, USA, 3rd edition. 72

Dutech, A. (1999). *Apprentissage d'environnements : approches cognitives et comportementales*. PhD thesis, Ecole Nationale Supérieure de l'Aéronautique et de l'Espace, Toulouse, France. 22, 33

Dutech, A. (2000). Solving POMDP using selected past-events. In *Proceedings of the 14th European Conference on Artificial Intelligence, ECAI2000*. 23, 34

Dutech, A., Aras, R., and Charpillet, F. (2006). Apprentissage par renforcement et théorie des jeux pour la coordination de systèmes multi-agents. In *Colloque Africain sur la Recherche en Informatique - CARI*, Cotonou, Bénin. 24

Dutech, A., Buffet, O., and Charpillet, F. (2001). Multi-agent systems by incremental gradient reinforcement learning. In *Proceedings of the seventeenth International Joint Conférence on Artificial Intelligence, IJCAI-01*, Seatle, USA. 24

Dutech, A. and Samuelides, M. (1999). Learning dynamical extensions of observation state in a partially observed environment. In *Workshop on Learning*, Snowbird, USA. 22

Dutech, A. and Samuelides, M. (2003). Apprentissage par renforcement pour les processus décisionnels de Markov partiellement observés. *Revue d'Intelligence Artificielle (RIA)*, 17(4) :559–589. 23, 34

Dutech, A. and Scherrer, B. (2001). Learning to use contextual information for solving partially observable Markov decision problems. In *Fifth European Workshop on Reinforcement Learning, EWRL-5*, Utrecht, Nederlands. 23

Fatès, N. (2010). Gathering agents on a lattice by coupling reaction-diffusion and chemotaxis. *To appear in Swarm Intelligence*. 78

Fleischer, J., Gally, J., Edelman, G., and Krichmar, J. (2007). Retrospective and prospective responses arising in a modeled hippocampus during maze navigation by a brain-based device. *Proc. of the National Academy of Sciences of the USA (PNAS)*, 104(4) :3556–3561. 69

Fudenberg, D. and Tirole, J. (1991). *Game Theory*. MIT Press. 42

Garcia, F. and Serre, F. (2000). Efficient asymptotic approximation in temporal difference learning. In *Proceedings of the 14th European Conference on Artificial Intelligence, ECAI2000*. 18

Gelly, S. and Wang, Y. (2006). Exploration exploitation in Go : UCT for Monte-Carlo GO. In *NIPS'06 Workshop on On-line Trading of Exploration and Exploitation*. 19

Ghavamzadeh, M. and Mahadevan, S. (2007). Hierarchical average-reward reinforcement learning. *Journal of Machine Learning Research*, 8 :2629–2669. 18, 37

Greenwald, A. and Hall, K. (2003). Correlated Q-learning. In *Proc. of the 20th Int. Conf. on Machine Learning (ICML)*. 49

Gregory, R. (1987). *The Oxford companion to the mind*. Oxford University Press, Oxford, UK. 8

Groupe PDMIA (2008). *Processus Décisionnels de Markov en Intelligence Artificielle. (Edité par Olivier Buffet et Olivier Sigaud)*, volume 1 & 2. Lavoisier - Hermes Science Publications. 13, 27, 31, 32, 44

Guestrin, C., Koller, D., and Parr, R. (2001). Solving factored POMDPs with linear value functions. In *Proc. of the IJCAI-01 Workshop on Planning under Uncertainty and Incomplete Information*, Seattle, WA. 30

Hansen, E. (1998). An improved policy iteration algorithm for partially observable MDPs. In *Advances in Neural Information Processing Systems 10 (NIPS)*. 30

Hansen, E., Bernstein, D., and Zilberstein, S. (2004). Dynamic programming for partially observable stochastic games. In *Proc. of the Nineteenth National Conference on Artificial Intelligence (AAAI-04)*. 46

Hansen, E. and Zilberstein, S. (2001). LAO* : a heuristic search algorithm that finds solution with loops. *Artificial Intelligence*, 129 :35–62. 19

Harnad, S. (1990). The symbol grounding problem. *Physica D*, 42 :335–346. 8

Hart, S. and Mas-Colell, A. (2003). Uncoupled dynamics do not lead to Nash equilibrium. *American Economic Review*, pages 1830–1836. 42, 49

Hasinoff, S. (2002). Reinforcement learning for problems with hidden state. Technical report, University of Toronto, Department of Computer Science. 31

Haugeland, J. (1985). *Artificial Intelligence : The Very Idea*. MIT Press, Cambridge. 7

Haugeland, J. (1989). *L'esprit dans la machine*. Odile Jacob, Paris. 7

Hendriks-Jensen, H. (1996). *Catching Ourselves in the Act*. The MIT Press, Cambridge, 2nd edition. 74

Hoey, J. and Little, J. (2004). Value directed learning of gestures and facial displays. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR'04)*, volume 02, pages 1026–1033. 33

Hu, J. and Wellman, M. (1998a). Multiagent reinforcement learning : theoretical framework and an algorithm. In *Proceedings of the Fifteenth International Conference on Machine Learning, ICML-98*, pages 242–250. 48

Hu, J. and Wellman, M. (1998b). Online learning about other agents in a dynamic multiagent system. In *Second International Conference on Autonomous Agents*, pages 239–246. 43

Jaakkola, T., Singh, S., and Jordan., M. (1994). On the convergence of stochastic iterative dynamic programming algorithms. *Neural Computation*, 6 :1186–1201. 31

Jaeger, H. (2000). Observable operator models for discrete stochastic time series. *Neural Computation*, 12(6) :1371–1398. 30

James, M. and Singh, S. (2004). Learning and discovery of predictive state representations in dynamical systems with reset. In *Proc. of the Twenty-first Int. Conf. of Machine Learning (ICML'04)*. 31

Jaulmes, R., Pineau, J., and Precup, D. (2005). Learning in non-stationary partially observable Markov decision processes. In *ECML Workshop on Reinforcement Learning in Non-Stationary Environments*, Porto, Portugal. 33

Jordan, M. (1999). *Learning in graphical models*. The MIT Press, Cambridge. 33

Kaelbling, L., Littman, M., and Cassandra, A. (1998). Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101 :99–134. 27

Kearns, M. and Koller, D. (1999). Efficient reinforcement learning in factored MDPs. In *Proc. of the 14th Int. Joint Conf. on Artificial Intelligence, IJCAI'99, Stockholm*. 18, 61

Kearns, M. and Singh, S. (2002). Near-optimal reinforcement learning in polyniomial time. *Machine Learning*, 49 :209–232. 18

Kimura, H., Miyazaki, K., and Kobayashi, K. (1997). Reinforcement learning in POMDPs with function approximation. In *Proc. of the Fourteenth Int. Conf. on Machine Learning (ICML'97)*, pages 152–160. 31

Krichmar, J., Seth, A., Nitz, D., Fleischer, J., and Edelman, G. (2005). Spatial navigation and causal analysis in a brain-based device modeling cortical-hippocampal interactions. *Neuroinformatics*, 3(3) :197–222. 69

Lanzi, P. (2000). Adaptive agents with reinforcement learning and internal memory. In *From Animals to Animats, Proc. of the Sixth Int. Conf. on the Simulation of Adaptive Behavior (SAB2000)*, Paris (France). 31

Laud, A. (2004). *Theory and application of reward shaping in reinforcement learning*. PhD thesis, University of Illinois at Urbana-Champaign. 52

Lee, C.-S., Wang, M.-H., Chaslot, G., Hoock, J.-B., Rimmel, A., Teytaud, O., Tsai, S.-R., Hsu, S.-C., and Hong, T.-P. (2009). The Computational Intelligence of MoGo Revealed in Taiwan's Computer Go Tournaments. *IEEE Transactions on Computational Intelligence and AI in games*. 19

Li, L., Littman, M., and Walsh, T. (2008). Knows what it knows : A framework for self-aware learning. In *Proc. of the Twenty-Fifth Int. Conf. on Machine Learning (ICML'08)*. 61

Li, L., Walsh, T., and Littman, M. (2006). Towards a unified theory of state abstraction for MDPs. In *Proc. of the Ninth Int. Symp. on AI & Math (AIMATH'06)*. 18

Lindsay, P. and Norman, D. (1977). *Human information processing : An introduction to psychology*. Academic Press, New York. 8

Littman, M. (1994a). Markov games as a framework for multi-agent reinforcement learning. In *Proc. of the eleventh Int. Conference on Machine Learning, San Fransisco, CA*. 48

Littman, M. (2001). Friend-or-foe Q-learning in general-sum games. In *Proc. of the 18th Int. Conf. on Machine Learning (ICML)*. 48

Littman, M., Sutton, R., and Singh, S. (2002). Predictive representation of state. In *Advances in Neural Information Processing Systems 14 (NIPS)*. 30

Littman, M. L. (1994b). Memoryless policies : Theoretical limitations and practical results. In *From Animals to Animats 3 : Proceedings of the Third International Conference on Simulation of Adaptive Behavior (SAB'94)*, Cambridge, MA. 31

Luce, R. and Raiffa, H. (1957). *Games and Decision : Introduction and Critical Survey*. Wiley. 4

Lungarella, M., Metta, G., Pfeifer, R., and Sandini, G. (2003). Developmental robotics : a survey. *Connection Science*, 15(4) :151–190. 35, 75

Martinez-Cantin, R., de Freitas, N., Brochu, E., Castellanos, J., and Doucet, A. (2009). A bayesian exploration-exploitation approach for optimal online sensing and planning with a visually guided mobile robot. *Autonomous Robot*, 27 :93–103. 67

Mataric, M. (1997). Learning social behavior. *Robotics and Autonomous System*, 20 :191–204. 56

McCallum, A. (1996). Learning to use selective attention and short-term memory in sequential tasks. In *From Animals to Animats, Proc. of the Fourth Int. Conf. on Simulating Adaptive Behavior*. 34

McCracken, P. and Bowling, M. H. (2005). Online discovery and learning of predictive state representations. In *Advances in Neural Information Processing Systems 18 (NIPS'05)*. 30, 31

Moore, A. and Atkeson, C. (1993). Memory based reinforcement efficient computation by prioritized sweeping. In Hanson, S., Giles, C., and Cowan, J., editors, *Advances in neural information processing systems*, volume 5. Morgan Kaufmann. 61

Munos, R. (2004). *Contributions à l'apprentissage par renforcement et au contrôle optimal avec approximation.* PhD thesis, Univ. Pierre et Marie Curie, Paris. 20

Munos, R. (2006). Geometric variance reduction in Markov chains. Application to value function and gradient estimation. *Journal of Machine Learning Research*, 7 :413–427. 20

Munos, R. (2007). Performance bounds in Lp norms for approximate value iteration. *SIAM Journal on Control and Optimization*, 46. 20

Myerson, R. (1991). *Game Theory : Analysis of Conflict.* Harvard University Press. 42

Neisser, U. (1967). *Cognitive Psychology.* Appleton Century Crofts, New York. 8

Ng, A., Harada, D., and Russell, S. (1999). Policy invariance under reward transformations : Theory and application to reward shaping. In *Proceedings of the Sixteenth International Conference on Machine Learning, ICML-99*, pages 278–287. 52

Nudelman, E., Wortman, J., Shoham, Y., and Leyton-Brown, K. (2004). Run the GAMUT : A comprehensive approach to evaluating game-theoretic algorithms. In *Proc. of the third Int. Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS'04)*, pages 880–887. 51

Oliehoek, F., Spaan, M., and Vlassis, N. (2008). Optimal and approximate Q-value functions for decentralized POMDPs. *Journal of Artificial Intelligence Research (JAIR)*, 32 :289–353. 44, 46

Osborne, M. J. and Rubinstein, A. (1994). *A Course in Game Theory.* The MIT Press, Cambridge, Mass. 24

Oudeyer, P.-Y. and Kaplan, F. (2006). Discovering communication. *Connection Science*, 18(2) :189–206. 56

Oudeyer, P.-Y., Kaplan, F., and Hafner, V. (2007). Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionnary Computation*, 11(2) :265–286. 38, 56, 69, 74

Papadimitriou, C. (1994). *Computational complexity.* Reading, MA :Adison-Wesley. 30

Parr, R. (1998). *Hierarchical control and learning for Markov decision processes.* PhD thesis, Computer Science, University of California at Berkeley. 18

Peng, J. and Williams, R. (1996). Incremental multi-step Q-learning. *Machine Learning*, 22 :283–290. 18

Peshkin, L. (2002). *Policy Search for Reinforcement Learning.* PhD thesis, Brown University. 17

Pfeifer, R. and Scheier, C. (2001). *Understanding Intelligence.* The MIT Press. 8, 9

Pineau, J., Gordon, G., and Thrun, S. (2003). Point-based value iteration : An anytime algorithm for POMDPs. In *Proc. of the Int. Joint Conf. on Artificial Intelligence (IJCAI'03)*, pages 1025 – 1032. 30

Price, B. and Boutilier, C. (1999). Implicit imitation in multiagent reinforcement learning. In *Proceedings of the Sixteenth International Conference on Machine Learning, ICML-99.* 56

Puterman, M. (1994). *Markov Decision Processes : discrete stochastic dynamic programming*. John Wiley & Sons, Inc. New York, NY. 4, 13, 15, 20

Rabiner, L. and Juang, B. (1986). An introduction to hidden Markov models. *IEEE ASSP Magazine*, 3(1). 32

Rachelson, E., Teichteil, F., and Garcia, F. (2007). XMDP : un modèle de planification temporelle dans l'incertain à actions paramétriques. In *Journées Françaises Planification Décision Apprentissage*. 20

Randlov, J. (2000). Shaping in reinforcement learning by changing the physics of the problem. In *Proceedings of the 17th International Conference on Machine Learning, (ICML'00)*, pages 767–774. 52

Randlov, J. and Alstrom, P. (1998). Learning to drive a bicycle using reinforcement learning and shaping. In *Proceedings of the 15th International Conference on Machine Learning, (ICML-98)*. 52

Riedmiller, M., Gabel, T., and Hafner, R. (2009). Reinforcement learning for robot soccer. *Autonomous Robot*, 27 :55–73. 67

Rougier, N. and Vitay, J. (2005). Emergence of attention within a neural population. *Neural Networks*. 58

Russell, S. and Norvig, P. (1995). *Artificial Intelligence : A modern approach*. Prentice Hall. 4

Seuken, S. and Zilberstein, S. (2007). Memory-bounded dynamic programming for DEC-POMDPs. In *Proc. of the Twentieth Int. Joint Conf. on Artificial Intelligence (IJCAI'07)*. 46

Shalizi, C. (2001). *Causal Architecture, Complexity and Self-Organization for Time Series and Cellular Automata*. PhD thesis, University of Wisconsin at Madison, Physics Department. 35

Shatkay, H. (1999). Learning hidden Markov models with geometrical constraints. In *Proc. of the Fifteenth Conf. on Uncertainty in Artificial Intelligence, (UAI'99)*, pages 602–611. 33

Siciliano, B. and Khatib, O., editors (2008). *Handbook of Robotics*. Springer Verlag, Berlin. 69

Simon, H. (1969). *The science of the artificial*. MIT Press. 9

Singh, S., Jaakkola, T., and Jordan, M. (1994). Learning without state estimation in partially observable markovian decision processes. In *Proceedings of the Eleventh International Conference on Machine Learning*. 31, 37

Singh, S., James, M. R., and Rudary, M. R. (2004). Predictive state representations : A new theory for modeling dynamical systems. In *Proc. of the twentieth Conf. on Uncertainty in Artificial Intelligence (UAI'04)*. 30

Singh, S., Littman, M., Jong, N., Pardoe, D., and Stone, P. (2003). Learning predictive state representations. In *Proc. of the Twentieth Int. Conf. of Machine Learning (ICML'03)*. 31

Skinner, B. (1953). *Science and Human Behavior*. Collier-MacMillan, New-York. 4, 51

Smart, W. (2002). *Making reinforcement learning work on real robots.* PhD thesis, Department of Computer Science, Brown University. 61

Smart, W. and Kaelbling, L. (2002). Effective reinforcement learning for mobile robots. In *Proc. of the International Conference on Robotic and Automation.* 61

Spaan, M. and Vlassis, N. (2005). Perseus : Randomized point-based value iteration for POMDPs. *Journal of Artificial Intelligence Research (JAIR)*, 24 :195–220. 30

Sporns, O. and Alexander, W. H. (2002). Neuromodulation and plasticity in an autonomous robot. *Neural Netw.*, 15(4) :761–774. 74, 77

Staddon, J. (1983). *Adaptive Behavior and Learning.* Cambridge University Press. 51

Stolcke, A. and Omohundro, S. (1992). Hidden Markov model induction by bayesian model merging. In *Advances in neural information processing systems*, volume 5. 33

Stolcke, A. and Omohundro, S. (1994). Best-first model merging for hidden Markov model induction. Technical Report TR-94-003, ICSI, Berkeley, CA. 33

Sutton, R. (1988). Learning to predict by the methods of temporal differences. *Machine Learning*, 3 :9–44. 16

Sutton, R. (1996). Generalization in reinforcement learning : Successful examples using sparse coarse coding. In *Advances in Neural Information Processing Systems 8 (NIPS)*, pages 1038–1044. MIT Press. 4, 13, 58

Sutton, R., McAllester, D., Singh, S., and Mansour, Y. (2000). Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems 12, NIPS00.* 18

Tesauro, G. (1995). Temporal difference learning and TD-gammon. *Communications of the ACM*, 38(3). 12

Theocharous, G., Rohanimanesh, K., and Mahadevan, S. (2001). Learning hierarchical partially observable Markov decision process models for robot navigation. In *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA'01).* 31

Thiébaux, S., Gretton, C., Slaney, J., Price, D., and Kabanza, F. (2006). Decision-theoretic planning with non-markovian rewards. *Journal of Artificial Intelligence Research (JAIR)*, 25 :17–74. 21

Thiery, C. (2007). Contrôle optimal stochastique et le jeu de tetris. Master's thesis, Université Henri Poincaré - Nancy I. 18

Thorndike, E. (1911). *Animal Intelligence.* MacMillan Company, New York. 4

Thrun, S. (1992). Efficient exploration in reinforcement learning. Technical Report CMU-CS-92-102, Computer Science Department, Carnegie Mellon University. 61

Thrun, S., Burgard, W., and Fox, D. (2005). *Probabilistic Robotics.* MIT Press, Cambridge, MA. 12

Turkewitz, G. and Kenny, P. (1982). Limitation on input as a basis for neural organization and perceptual development : a preliminary theoretical statement. *Developmental Psychology*, 15 :357–368. 74

Varela, F. (1989). *Connaître les sciences cognitives. Tendances et perspectives (1988).* Ed. du Seuil, Paris. 3

Vlassis, N., Toussaint, M., Kontes, G., and Piperidis, S. (2009). Learning model-free robot control by Monte Carlo EM. *Autonomous Robot*, 27 :123–130. 67, 69

Watkins, C. (1989). *Learning from delayed rewards.* PhD thesis, King's College of Cambridge, UK. 4, 16

Weiss, G. and Sen, S. (1996). *Adaptation and learning in multi-agent systems*, volume 1042. Springer-Verlag, Lectures Notes in Artificial Intelligence. 39

Weng, J., McClelland, J., Pentland, A., Sporns, O., Stockman, I., Sur, M., and Thelen, E. (2000). Autonomous mental development by robots and animals. *Science*, 291(5504) :599–600. 38

Wiering, M. and Schmidhuber, J. (1996). HQ-learning : discovering markovian subgoals for non-markovian reinforcement learning. Technical Report IDSIA-95-96, Istituto Dalle Molle di Studi sull'Intelligenza Artificiale, Switzerland. 37

Wiering, M. and Schmidhuber, J. (1997). HQ-learning. *Machine Learning*, 6(2). 18

Williams, R. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8 :229–256. 17

Wingate, D. and Singh, S. (2007). On discovery and learning of models with predictive state representations of state for agents with continuous actions and observations. In *Proc.of the Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS'07).* 30

Younes, H. and Simmons, R. (2004). Solving generalized semi-Markov decision processes using continuous phase-type distributions. In *Proc. of the Nat. Conf. on Artificial Intelligence (AAAI).* 20

Zang, N. and Lio, W. (1996). Planning in stochastic domains : Problem characteristics and approximation. Technical report, Tech. report HKUST-CS96-31, Honk-Kong University of Science and Technology. 30